

Trabajo Fin de Máster
Máster en Electrónica, Tratamiento de Señal y
Comunicaciones

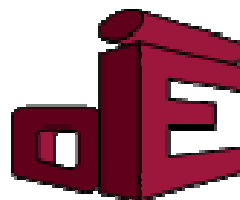
Diseño y Desarrollo de un Sistema de Control para
Pantallas LED Modulares

Autor: José Juan Bautista Pulido

Tutores: Hipólito Guzmán Miranda
Fernando Muñoz Chavero

Dep. Ingeniería Electrónica
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2017



Trabajo Fin de Máster
Máster en Electrónica, Tratamiento de Señal y Comunicaciones

Diseño y Desarrollo de un Sistema de Control para Pantallas LED Modulares

Autor:

José Juan Bautista Pulido

Tutores:

Hipólito Guzmán Miranda

Profesor contratado Doctor

Fernando Muñoz Chavero

Profesor titular

Dep. Ingeniería Electrónica
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2017

Trabajo Fin de Máster: Diseño y Desarrollo de un Sistema de Control para Pantallas LED Modulares

Autor: José Juan Bautista Pulido

Tutores: Hipólito Guzmán Miranda
Fernando Muñoz Chavero

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2017

El Secretario del Tribunal

A mi familia

A mis maestros

Resumen

El Trabajo Fin de Máster consiste en el diseño y desarrollo de un sistema de control para pantallas LED modulares. El sistema permite la realización de cualquier tipo de test sobre los módulos LED (unidad básica con la que se construyen las pantallas) tanto en laboratorio de I+D, fábrica o taller de reparaciones. El sistema también sirve como prototipo para un sistema de control de pantallas práctico para imágenes estáticas y de baja velocidad de actualización. El sistema está desarrollado a partir de una FPGA para que pueda ser flexible y fácilmente escalable.

Resumen	ix
Índice	xi
1 Introducción	13
2 Objetivos y alcance	25
3 Introducción al control de pantallas LED modulares	11
3.1 <i>Matriz LED</i>	11
3.1.1 Introducción	11
3.1.2 Estructura básica de una matriz LED	11
3.1.3 Multiplexación de una matriz LED	12
3.1.4 Estructura básica de un sistema de control	13
3.1.5 Control del brillo a través de modulación de ancho de pulso (PWM)	13
3.1.6 Cuadros (Frames)	14
3.2 <i>Sistema de pantalla LED práctico</i>	15
3.2.1 Estructura del sistema de pantalla LED	15
3.2.2 El módulo LED	16
3.2.2.1 Corrección gamma	20
3.2.2.2 Transformación del espacio de color	21
3.2.2.3 Distribución de datos	21
4 Caracterización y funcionamiento de un módulo LED	23
4.1 <i>Descripción</i>	23
4.2 <i>Funcionamiento</i>	23
5 Implementación de un controlador para módulo de pantalla LED basado en FPGA	29
5.1 <i>Introducción</i>	29
5.2 <i>Objetivos de diseño</i>	29
5.3 <i>Hardware del sistema</i>	30
5.3.1 Módulo de evaluación para FPGA	30
5.3.2 Diseño y desarrollo del hardware del controlador	33
5.3.2.1 Módulo 'ethernet'	34
5.3.2.2 Módulo 'rx_control'	38
5.3.2.3 Módulo 'ram_memory'	47
5.3.2.4 Módulo 'display_control'	48
5.3.2.5 Módulo 'dcm1'	54
6 Aplicación para PC para el envío de información al controlador del módulo LED (FPGA)	59
7 Pruebas y test del sistema	63
8 Conclusiones y trabajos futuros	71
9 Referencias	73

1 INTRODUCCIÓN

La mejor forma de dar una introducción a las Pantallas de LED Modulares es mostrar algunos ejemplos de sistemas prácticos y a partir de ellos empezar a descomponer el sistema en cada una de las partes que lo componen al mismo tiempo que se describen.

Se proponen como ejemplos las Pantallas de LED Modulares de la empresa Alfalite.

Desde 2016 Alfalite diseña, produce, distribuye y exporta pantallas LED para aplicaciones diversas, ofreciendo una amplia gama de productos tanto para instalaciones fijas como no permanentes, que pueden ser utilizados en sectores tan dispares como el entretenimiento, la publicidad o la comunicación.

La sede de Alfalite se encuentra en Rociana del Condado (Huelva - España) y cuenta con 2.000 m² de instalaciones en las que se llevan a cabo la investigación, diseño, pruebas, showroom y servicio de ventas.

Alfalite tiene su propio departamento de proyectos e instalaciones, que permite la organización de proyectos llave en mano o de cualquier otro tipo, independientemente de su complejidad. Los ingenieros de Alfalite han desarrollado diseños exclusivos que se han mejorado a lo largo del tiempo con gran éxito, como por ejemplo:

- Serie Alfapix: Especialmente diseñada para alquiler. Se caracteriza por su versatilidad, es ligera aunque robusta y sirve tanto para interiores como exteriores. Puede situarse en el suelo, apilada o colgada de una estructura. Este modelo puede tener Pixel Pitch de 6, 8, 10 y 15 mm. Destacar que supera los 6000 nits de brillo y tiene una velocidad de refresco superior 1920 Hz. Si se usa como suelo puede soportar hasta 250 Kg. por panel. El tamaño de los paneles es de 500 mm x 500 mm.
- Serie Broadcast: Pantallas HD para platós de TV, salas de conferencias, teatros, etc. Esta serie ha sido concebida para cumplir las notables exigencias de los platós de TV y del “directo” donde se requieren las más altas prestaciones. Sus principales características son una sorprendente profundidad de color, un alto contraste y uniformidad de brillo, una elevadísima velocidad de refresco (hasta los 3.820 Hz), un bajísimo nivel de ruido (no usan ventiladores), fiabilidad, fácil mantenimiento y bajo consumo. Usa un Pixel Pitch reducido (3.75, 4, 4.615, 5 y 6 mm) que permite alcanzar una óptima calidad de la imagen y altas resoluciones. Este Pixel Pitch hace que esta serie sea muy idónea para la construcción de videowalls de alta resolución (sin las molestas barras de separación de los clásicos videowalls LCD).

Como ejemplo de sistema práctico se muestran las pantallas utilizadas en la XXI Edición de los Premios Cadena Dial, que tuvo lugar en Tenerife, y donde se rindió homenaje a los talentos más destacados del panorama musical en español en el último año como Alejandro Sanz, David Bisbal o Raphael. La gala contó con el espectacular despliegue técnico de Sonoplus, con la puesta en escena de más de 200 metros cuadrados de paneles LED de Alfalite.

Otro ejemplo de sistema práctico es el de la pantalla utilizada en la Gala Inocente Inocente de RTVE de diciembre de 2014, donde estuvieron presente las pantallas LED de Alfalite de la serie Micropix 3mm (concebida para TV, velocidad de refresco por encima de 3840 Hz, escala de color de 16 bit, alto contraste y totalmente silenciosa).

A continuación se muestran algunas imágenes de las pantallas utilizadas en los eventos.



XXI Edición de los Premios Cadena Dial: Vista lateral.



XXI Edición de los Premios Cadena Dial: Vista frontal.



Gala Inocente Inocente de RTVE.

Las pantallas están compuestas de paneles como el que se muestra a continuación.



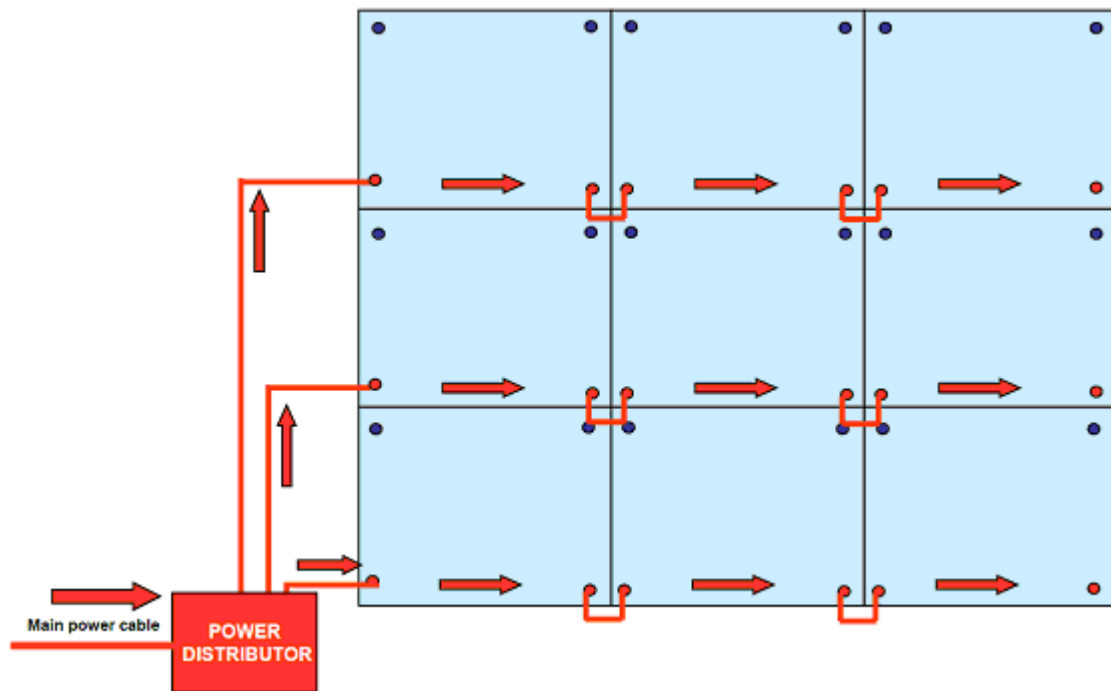
Panel LED de 500 mm x 500 mm.

Los paneles se ensamblan tal y como se muestra en la imagen que sigue para así dar la forma y tamaño a la pantalla.

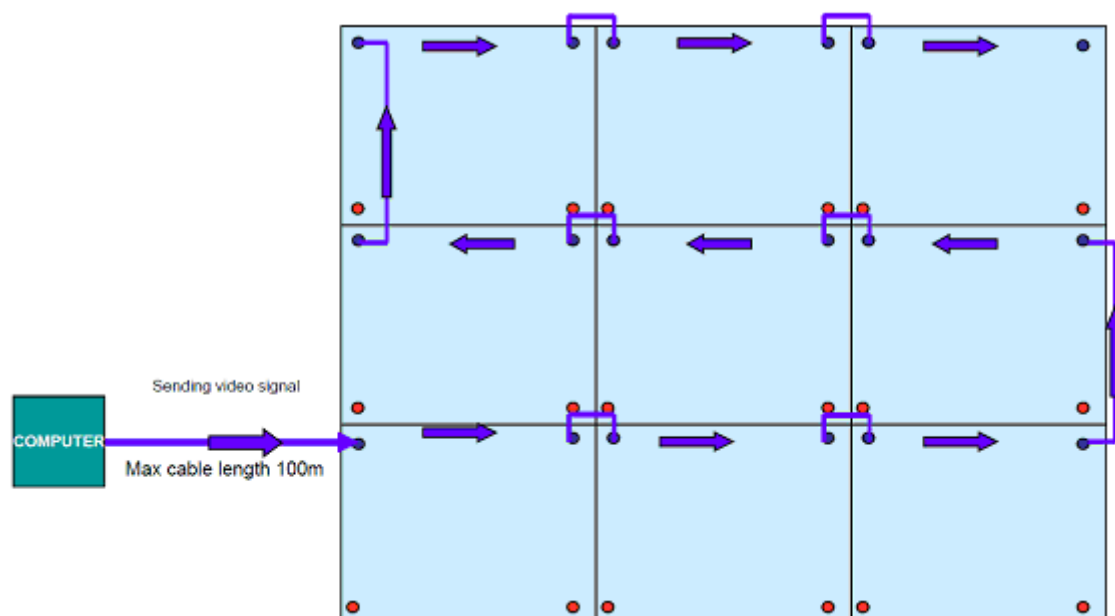


Ensamblado de paneles LED para construir la pantalla LED.

Una vez ensamblados los paneles estos se cablean tal y como se muestra en las siguientes figuras.



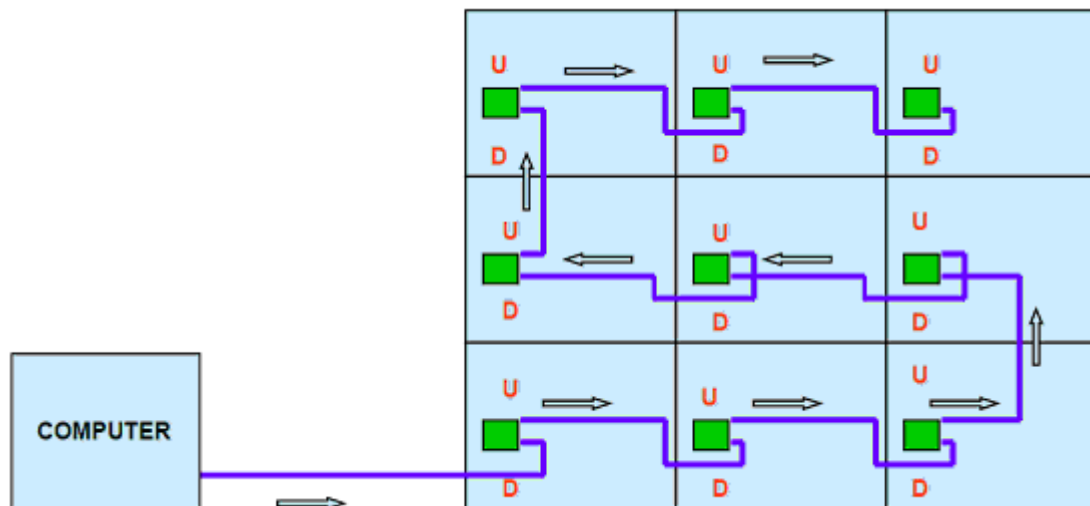
Conexión de cables de alimentación.



Conexión de cables de señal.

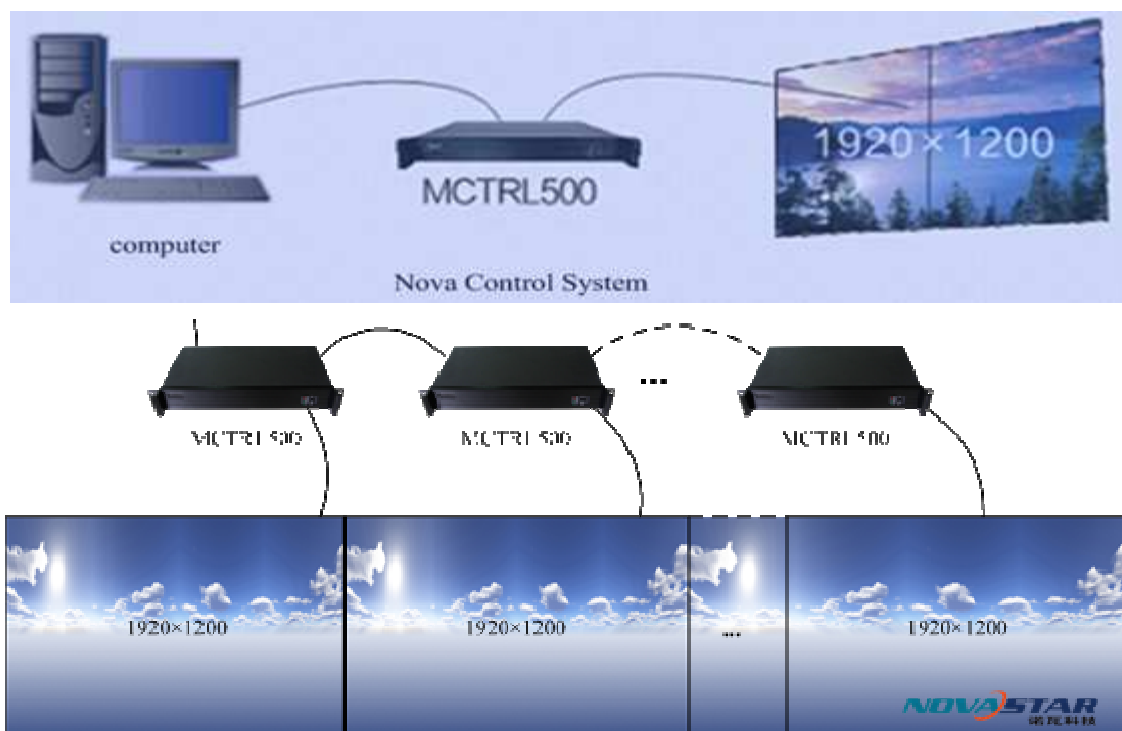
A continuación se realiza el direccionamiento de los paneles, lo cual nos permite "enseñar" a las tarjetas transmisoras y receptoras como distribuir la señal para mostrar una imagen grande en una pantalla completa que tiene una estructura modular.

Cada panel lleva instalado en su interior una tarjeta receptora. La imagen que sigue muestra como la señal de datos viaja desde el controlador o tarjeta transmisora al primer panel LED y como los cables de señal están conectados a las tarjetas receptoras del resto de paneles.

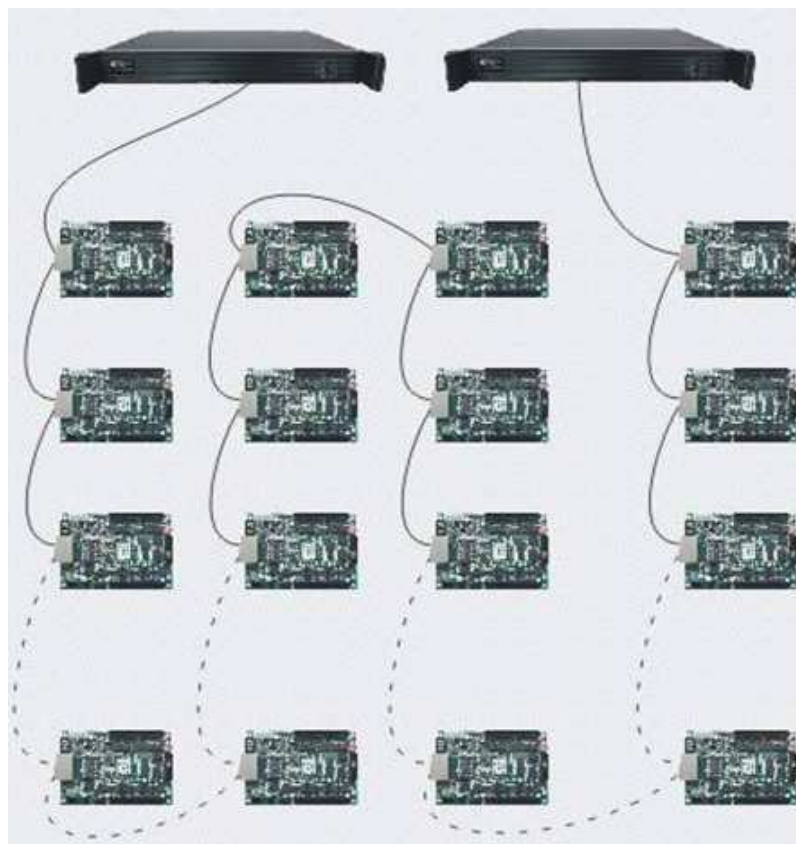
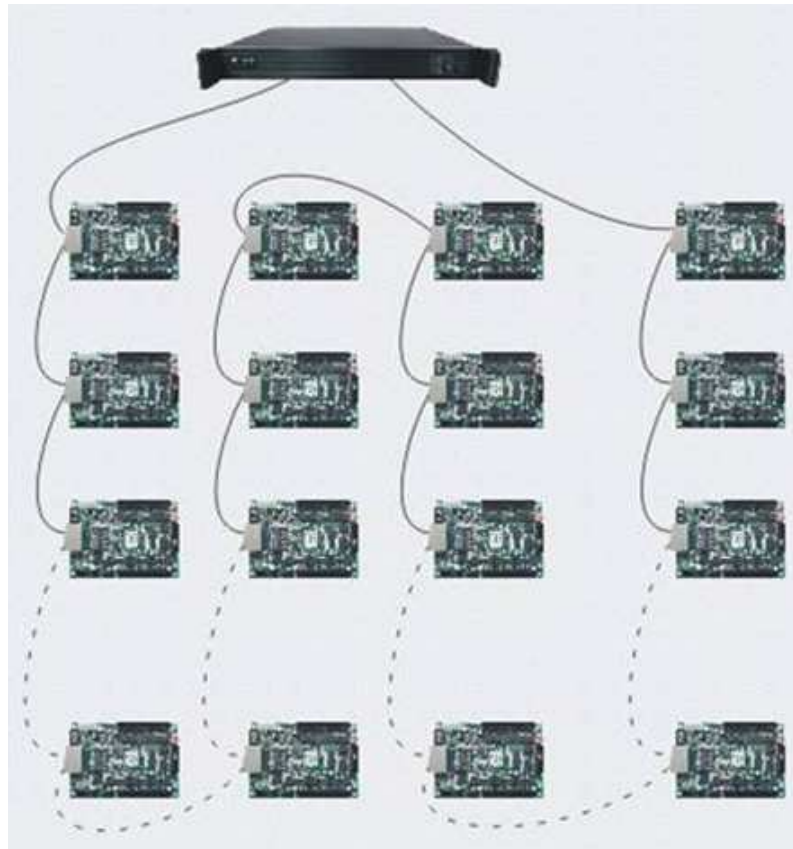


Direccionamiento de los paneles LED.

Si la pantalla es demasiado grande hay que recurrir al empleo de varios controladores en cascada tal y como se muestra en las figuras que siguen.



Controlador simple y en cascada.



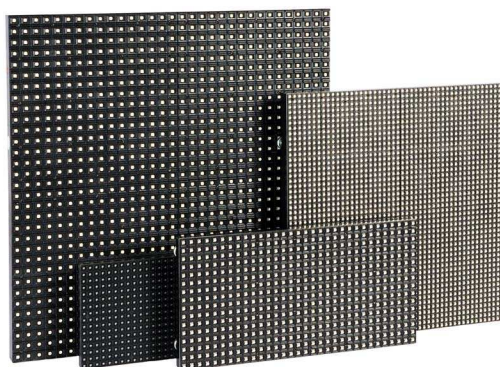
Controlador simple y en cascada.

Los paneles a su vez están compuestos de módulos. El número de módulos dependerá de la forma y tamaño de estos, siendo las configuraciones más típicas las de 4 módulos cuadrados (por ejemplo 64 x 64 pixeles) u 8 módulos rectangulares (por ejemplo 64 x 32 pixeles).

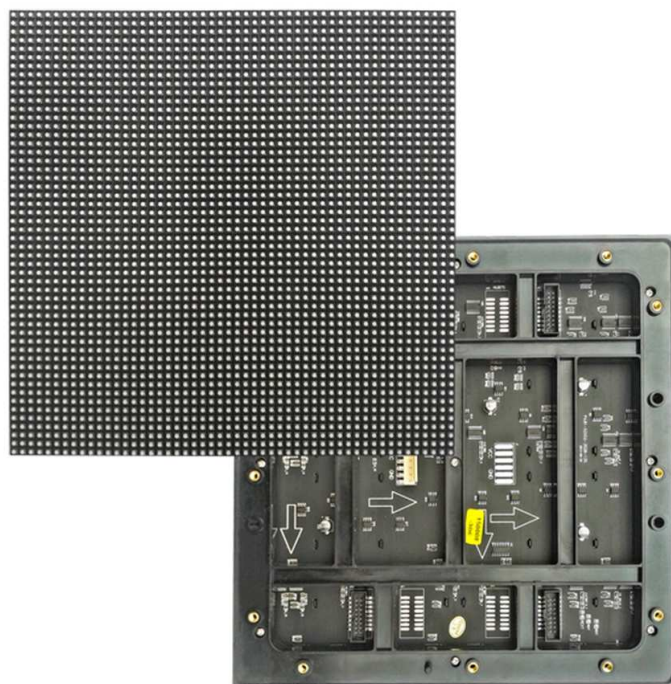


Modelo	Micropix 3 Standard
Pixel Pitch	3.91
Configuración LED	SMD RGB 3 en 1
Densidad de pixel	65.536 /m ²
Resolución panel	128x128
Peso panel	9.8 Kg
Brillo	1200 ~ 1500 nits
Ángulo de visión óptimo	160°/160°
Tasa de refresco	1680 ~ 3840 Hz
Rango de temperatura	-20° ~ +45°
Rango de humedad	10% ~ 90%
Ciclo de vida	100.000h
Cierres	-
Función MOM	-

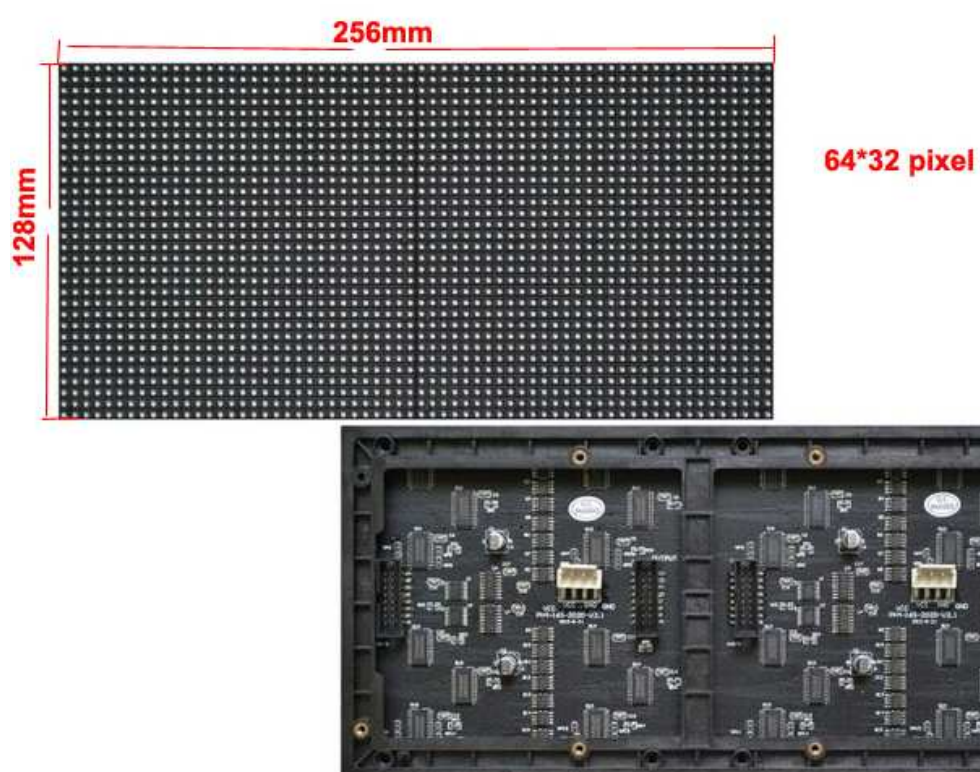
Frontal del panel donde pueden verse lo módulos LED.



Ejemplos de módulos LED.



Módulo LED de 64 x 64 pixeles.



Detalle de módulo LED de 64 x 32 pixeles.

Por último se muestran unos ejemplos de controlador y tarjeta receptora de la empresa Novastar de los más utilizados en el control de las pantallas.

Controlador Novastar M3 MCTRL660

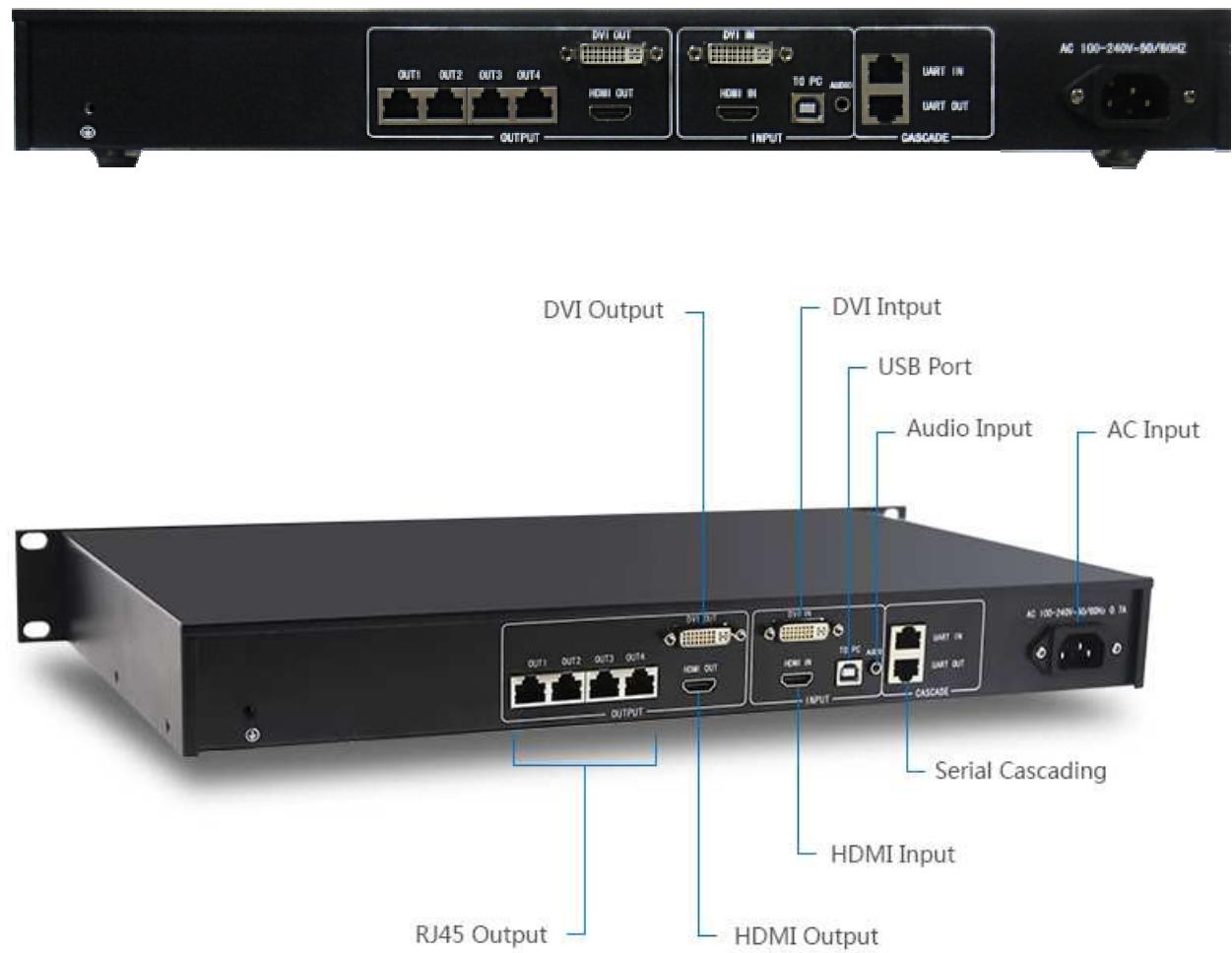
1. Características



MCTRL660 es el último controlador maestro independiente de NovaStar, destinado principalmente para el servicio de alquiler de pantallas y con las siguientes características:

- Adopta una arquitectura innovadora para implementar una configuración inteligente; la configuración de la pantalla se puede completar en 30 segundos.
- Adopta el motor Nova G4; la pantalla es estable y sin parpadeos libre de líneas de exploración; las imágenes son exquisitas y tienen un buen sentido de profundidad.
- Admite la nueva tecnología de corrección punto por punto de Nova; la corrección es rápida y eficiente.
- Puede implementar la calibración del balance de blancos y el mapeado gamma de color basado en función de las diferentes características de los LED utilizados por las pantallas para garantizar la reproducción de colores reales.
- Es el único sistema de control que admite entrada de interfaz multimedia de alta definición de 12 bits (HDMI) y la protección de contenido digital de gran ancho de banda (HDCP) en China.
- Admite la configuración de pantalla en cualquier momento sin PC.
- Admite el ajuste manual del brillo de la pantalla, que es conveniente y eficiente.
- Entrada HDMI/DVI.
- Entrada de audio HDMI.
- Fuente de vídeo HD de 12bit/10bit/8bit.
- Resoluciones soportadas: 2048×1152, 1920×1200, 2560×960, 1440×900 (12bit/10bit).
- Interfaz para sensor de luz.
- Soporta cascada.
- Presentación y procesamiento de escala de grises de 18bit.
- Formatos de vídeo: RGB, YCrCb4:2:2, YCrCb4:4:4.
- Interfaz USB para control.
- Interfaz de salida con 4 puertos Ethernet.

2. Interfaz



Tarjeta Receptora Novastar MRV220

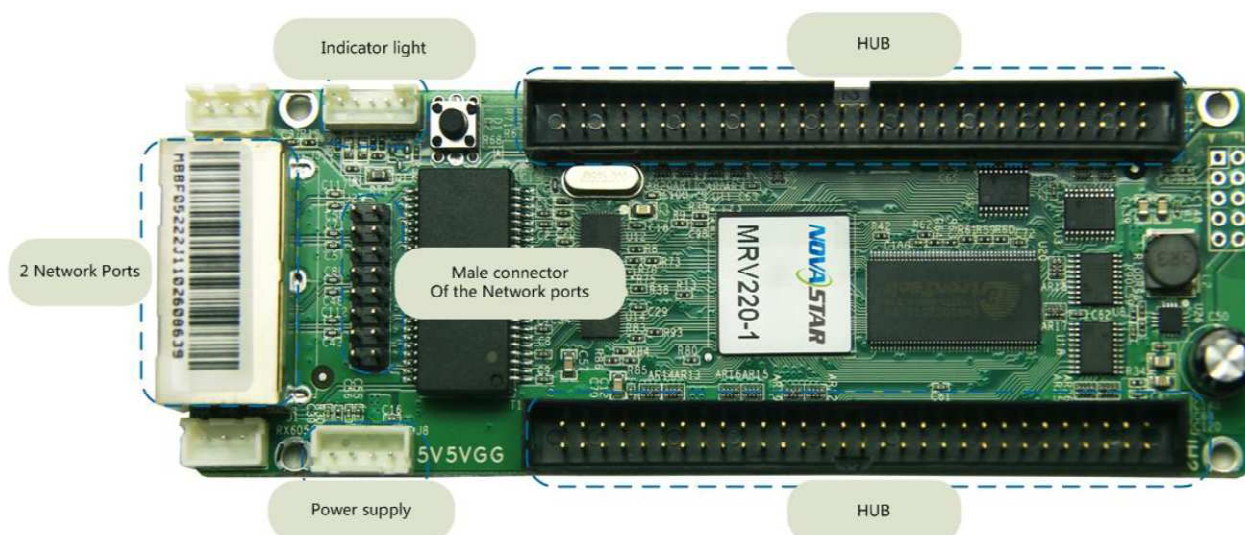
1. Características



MRV220 es la tarjeta receptora de Nova con hasta 28 grupos de datos RGB y las siguientes características:

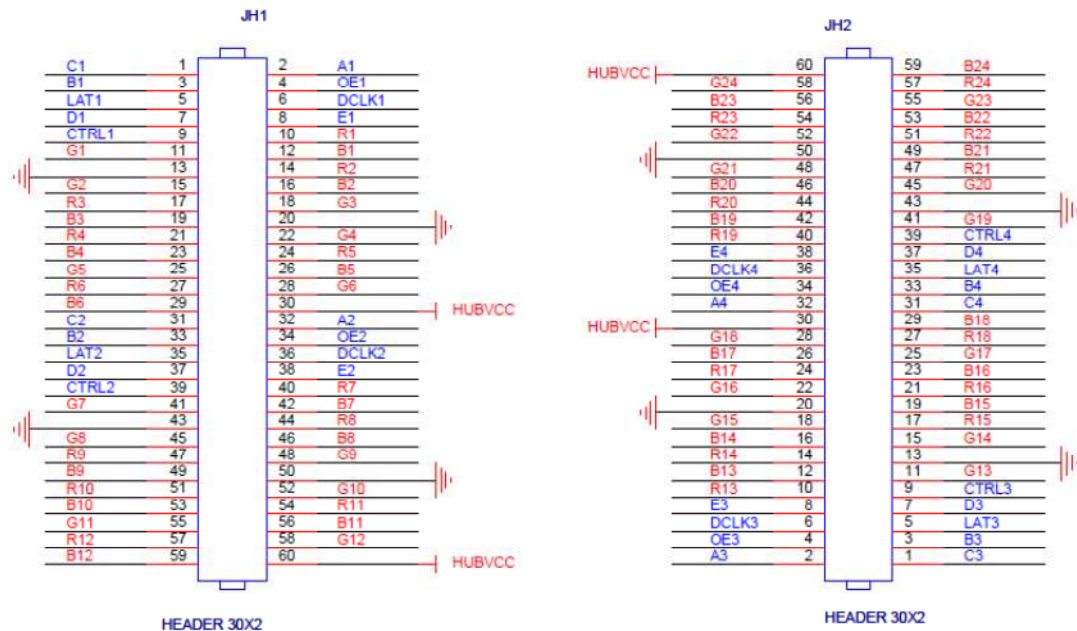
- Salidas para 24 o 28 grupos de datos RGB.
- Soporta resolución de 256x226.
- Recarga del archivo de configuración.
- Monitoreo de temperatura.
- Detección del estado del cable de comunicación Ethernet.
- Detección del voltaje de alimentación.
- Diferentes modos de refresco (High gray-scale, high-refresh, high and low brightness).
- Calibración cromática y de brillo Pixel por Pixel, coeficientes de calibración cromática y de brillo para cada LED.
- Cumple con los estándares RoHS y EU CE-EMC.

2. Interfaz

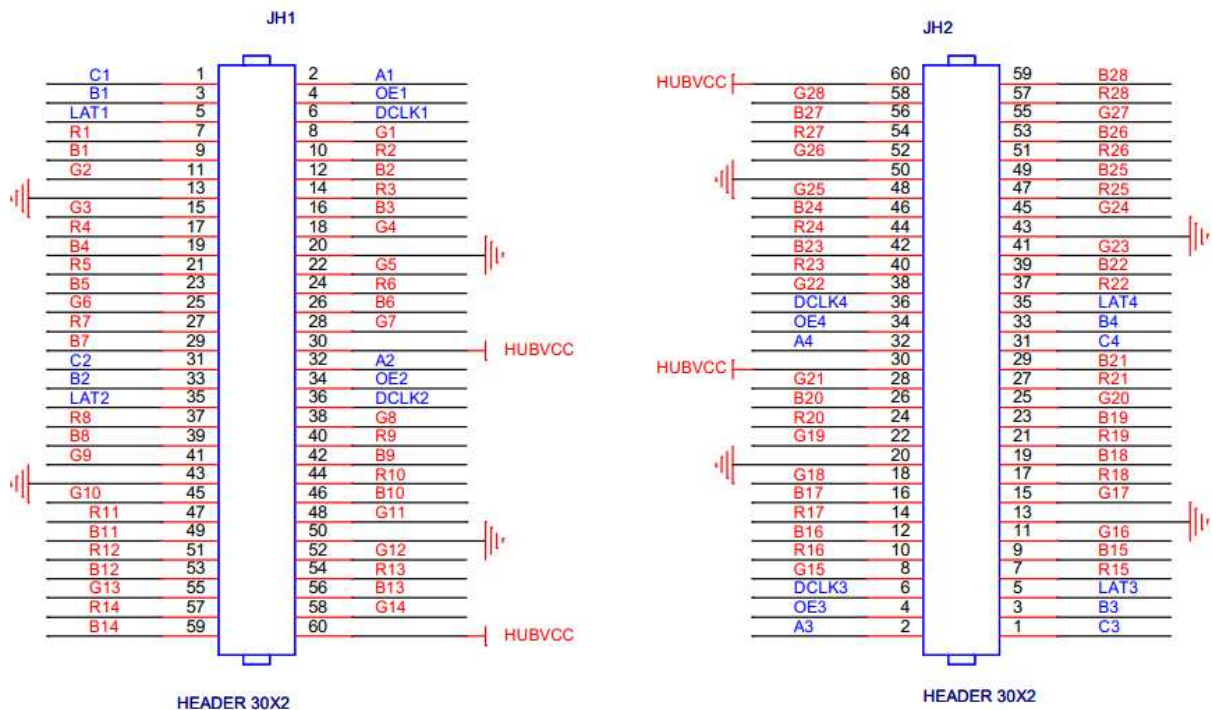


Con dos interfaces de 60 patillas admite dos modos de trabajo:

- Modo de 24 grupos de datos y 4 señales para la multiplexación de las líneas.



- Modo de 28 grupos de datos y 3 señales para la multiplexación de las líneas.



2 OBJETIVOS Y ALCANCE

Para el test de los módulos LED se emplean sistemas de control compuestos por un controlador y una tarjeta receptora como los mostrados en el capítulo 1, ya sea:

- En el laboratorio, para comprobar su funcionamiento y sus prestaciones, y para realizar la validación antes de ordenar su puesta en producción.
- En la fábrica, tras salir de la línea de montaje.
- En el taller, al finalizar su reparación.

El sistema de control permite la realización de test, pero no de cualquier test, ni tampoco permite un control de bajo nivel de las señales. Por estos motivos, como principal objetivo se tiene la obtención de un sistema que permita la realización de cualquier test que los ingenieros sean capaces de imaginar y programar y que permita un control total de las señales al más bajo nivel. Como objetivo secundario se pretende que el sistema que se desarrolle sirva como prototipo de un sistema de bajo coste para imágenes estáticas o de baja velocidad de actualización.

El alcance se fija en el control de un solo módulo LED, a ser posible con un sencillo módulo de evaluación para FPGA que hay disponible en el Departamento de Ingeniería Electrónica, debiéndose poder alcanzar al menos las siguientes funcionalidades y prestaciones:

- Como fuente de información se desarrollará una aplicación para PC que permita el envío de patrones de prueba, imágenes y vídeo de baja velocidad.
- El envío de la información desde el PC se realizará a través de una interfaz Ethernet.
- El controlador deberá disponer de una función para la corrección Gamma.
- Se deberá alcanzar una escala de grises de 256 niveles.
- La pantalla deberá actualizarse 60 veces por segundo.

3 INTRODUCCIÓN AL CONTROL DE PANTALLAS LED MODULARES

3.1 Matriz LED

3.1.1 Introducción

Los LEDs son dispositivos accionados por corriente. Es relativamente simple controlar varios LEDs individualmente, sin embargo, cuando el número de LEDs aumenta, la cantidad de recursos necesarios para operar estos LEDs crece a un nivel inmanejable. Como tales, los LEDs se arreglan a menudo en matrices para hacer un uso eficiente de los recursos.

En un formato de matriz, los LEDs se organizan en filas y columnas, lo cual exige que los LEDs se controlen mediante multiplexación. La secuencia de multiplexación inevitablemente requiere un procesamiento más complejo, pero es más eficiente en comparación con la conducción individual de cada LED. Mediante multiplexación también se puede controlar el brillo de cada LED de forma individual. Esto implica dividir la secuencia de conducción del LED en tres niveles en el dominio del tiempo.

3.1.2 Estructura básica de una matriz LED

Tomando como ejemplo una matriz 4x4, como se muestra a continuación en la Figura 3.1, puede verse como cada LED se puede direccionar especificando su ubicación en términos de filas y columnas. Por ejemplo, el LED superior izquierdo se direcciona como (A, 1) es decir, fila A, columna 1. Este método de direccionamiento también indica el flujo de corriente eléctrica. Para encender el LED (A, 1), la corriente circula desde A a 1. Si los interruptores están conectados a cada puerto A a D y 1 a 4, entonces, para encender el LED superior izquierdo, se hacen conducir los interruptores A y 1. Por los otros LEDs no circulará ninguna corriente debido a que sus interruptores de fila y columna no están conduciendo.

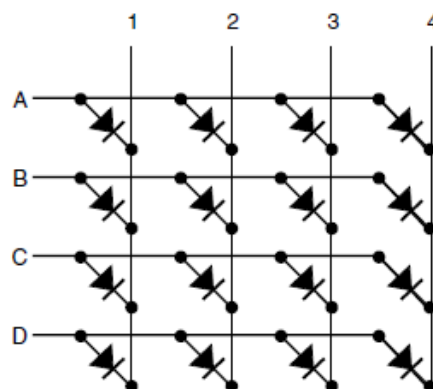


Figura 3.1. Arreglo matricial para fila de ánodo común.

3.1.3 Multiplexación de una matriz LED

La multiplexación es la técnica empleada para operar matrices de LED. Al multiplexar, sólo una fila de la matriz LED se activa en un momento dado, lo cual se requiere porque un extremo del LED está vinculado a una sola fila. En la Figura 3.2 puede verse que si la corriente se aplica a ambas filas A y B al mismo tiempo se hace imposible direccionar un LED individualmente dentro de esas dos filas.

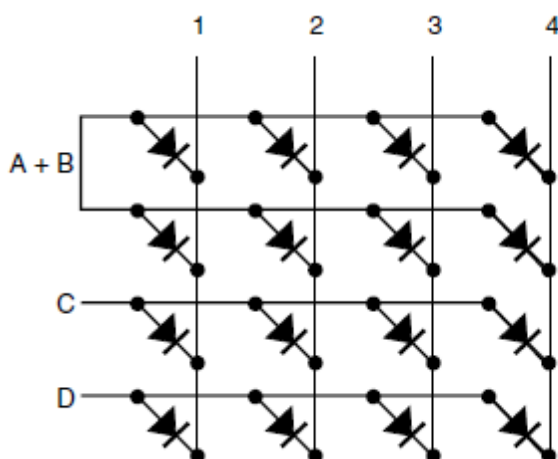


Figura 3.2. Si se hace conducir la línea 1 cuando A y B conducen se encenderán simultáneamente dos LEDs.

En la Figura 3.3 se muestra la secuencia de escalera (A a D) que se emplea en la multiplexación por división de tiempo. Sólo una fila se energiza en un periodo determinado, durante el cual los LEDs deseados se iluminan energizando las columnas apropiadas. Este proceso es conocido como escaneo.

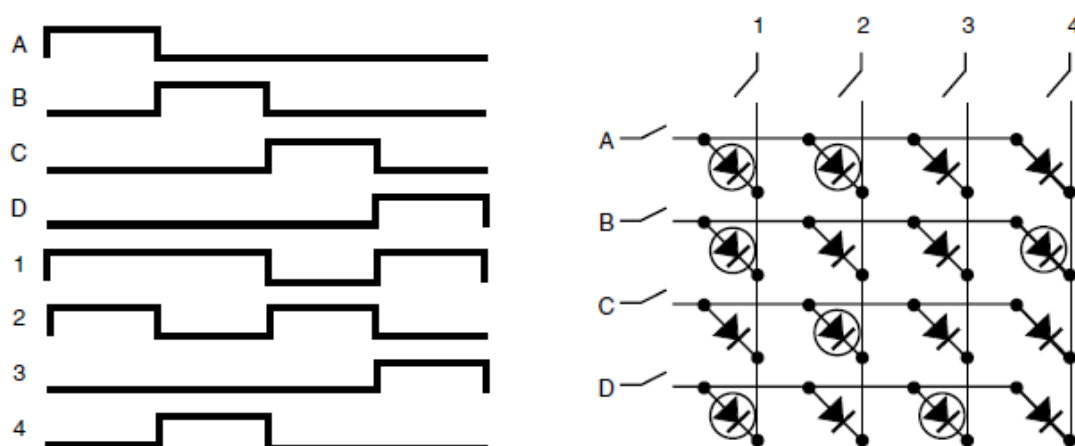


Figura 3.3. Multiplexación de una matriz LED. Los círculos en la figura de la derecha indican que LEDs están encendidos en la secuencia mostrada en la figura izquierda.

3.1.4 Estructura básica de un sistema de control

La Figura 3.4 sólo muestra una sección de la matriz. El esquema de control puede extenderse a matrices de LEDs muy grandes. El tamaño máximo depende de la velocidad máxima a la que la electrónica puede distribuir y procesar los datos. Para una configuración de fila de ánodo común, el sistema de control necesita sumideros de corriente constante en los drivers del lado de abajo.

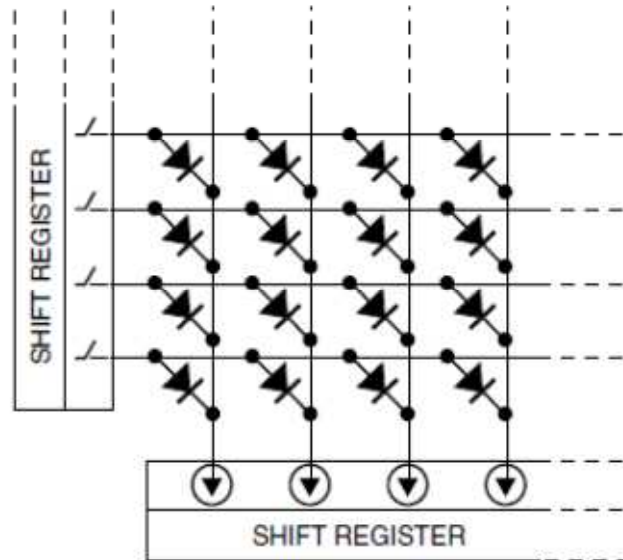


Figura 3.4. En el lado de arriba se utilizan interruptores electrónicos mientras que en el lado de abajo se utilizan sumideros de corriente. Se emplean registros de desplazamiento para aceptar la secuencia de conmutación en formato digital.

3.1.5 Control del brillo a través de modulación de ancho de pulso (PWM)

Sabemos que la salida de luz de un LED depende de la corriente que fluye a su través. Sin embargo, no es un método recomendable para controlar el brillo porque necesitará una fuente de corriente muy precisa. La técnica preferida para el control del brillo es la modulación de ancho de pulso (PWM) tal y como se muestra en la Figura 3.5.

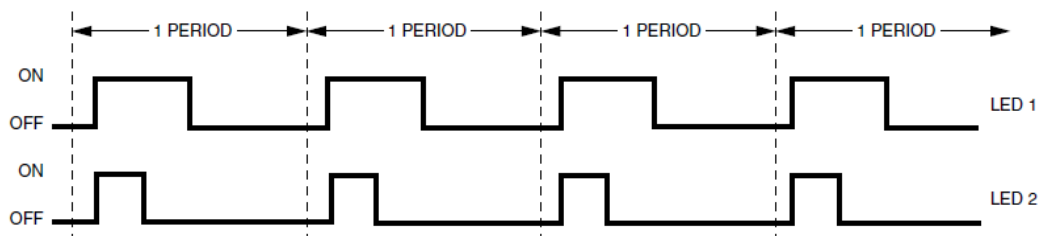


Figura 3.5. El LED1 brillará más que el LED2 porque está encendido más tiempo durante un periodo.

Como el sistema de control activará una fila entera al mismo tiempo, para controlar el brillo de cada LED individualmente hay que dividir cada período de escaneo en intervalos de tiempo, dando lugar a una jerarquía de dominio de tiempo.

La técnica PWM que se muestra en la Figura 3.6 se puede extender más allá del sistema de cuatro niveles de grises, cuanto más estrecho sea el intervalo de tiempo, más fino es el control de brillo.

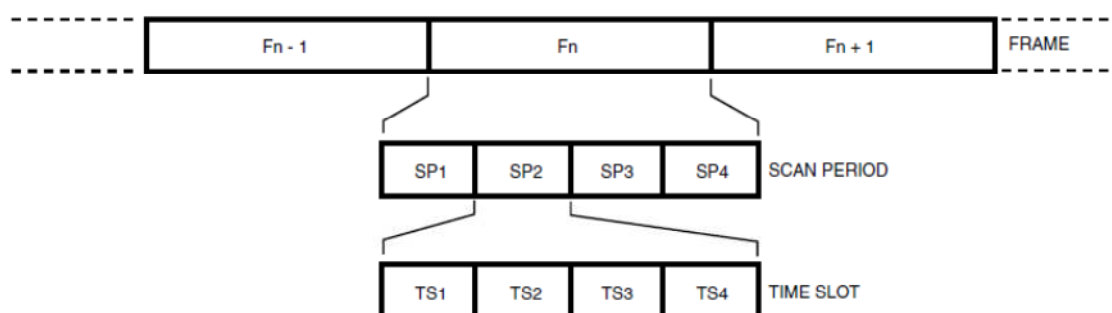


Figura 3.6. Jerarquía para un sistema de cuatro niveles de grises, ciclo de trabajo de 1/4.

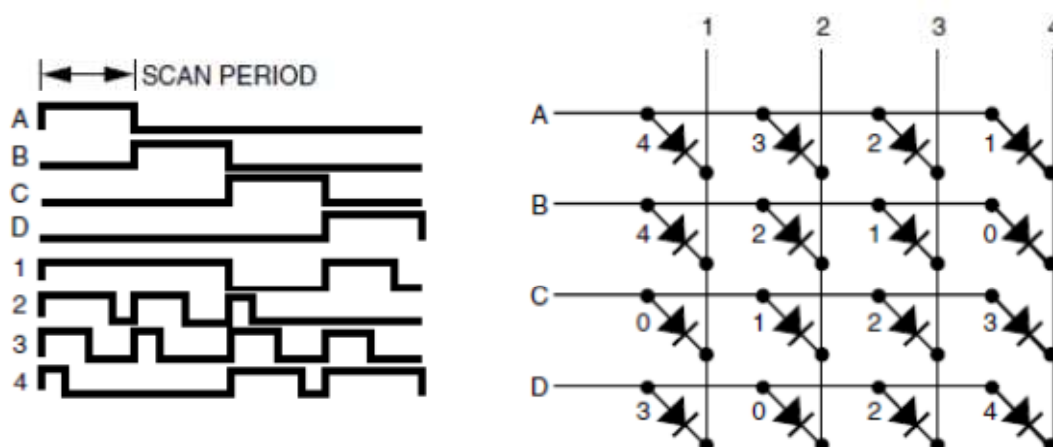


Figura 3.7. Técnica de control de brillo de LED individual mediante multiplexación. El diagrama de tiempos de la izquierda muestra cuatro intervalos de tiempo dentro de cada periodo de scaneo. Cada periodo de scaneo se corresponde con una fila de LEDs. La figura de la derecha muestra el brillo relativo de cada LED, el brillo decrece desde 4 a 0.

3.1.6 Cuadros (Frames)

Los cuadros se definen como la imagen final en la pantalla que debe ser presentada al observador. El video funciona presentando un conjunto de cuadros tan rápido que el observador no percibe ninguna discontinuidad. La velocidad a la que los cuadros se refrescan se denomina frecuencia de actualización. Si la frecuencia está por encima de cierta frecuencia umbral, el observador no notará ningún parpadeo. Para pantallas LED, se recomienda una frecuencia de actualización por encima de 60 Hz.

3.2 Sistema de pantalla LED práctico

3.2.1 Estructura del sistema de pantalla LED

La fuente de señal proporciona los datos de imagen (DVD o PC). Dividir la pantalla en varios componentes permite una gestión más fácil de los datos. Cada nivel contiene lógica para manejar la distribución de datos y generación de señales de control. En el nivel superior se encuentra el controlador de la pantalla, seguido por el controlador del panel que gobierna la operación del módulo LED. Una configuración típica es un módulo LED de 64x64 píxeles. Cada panel contiene 2x2 módulos, que dan como resultado un panel LED de 128x128 píxeles. Por último, una pantalla como la de la figura 3.8 puede contener 4x3 paneles, produciendo una pantalla LED de 512x 384 píxeles.

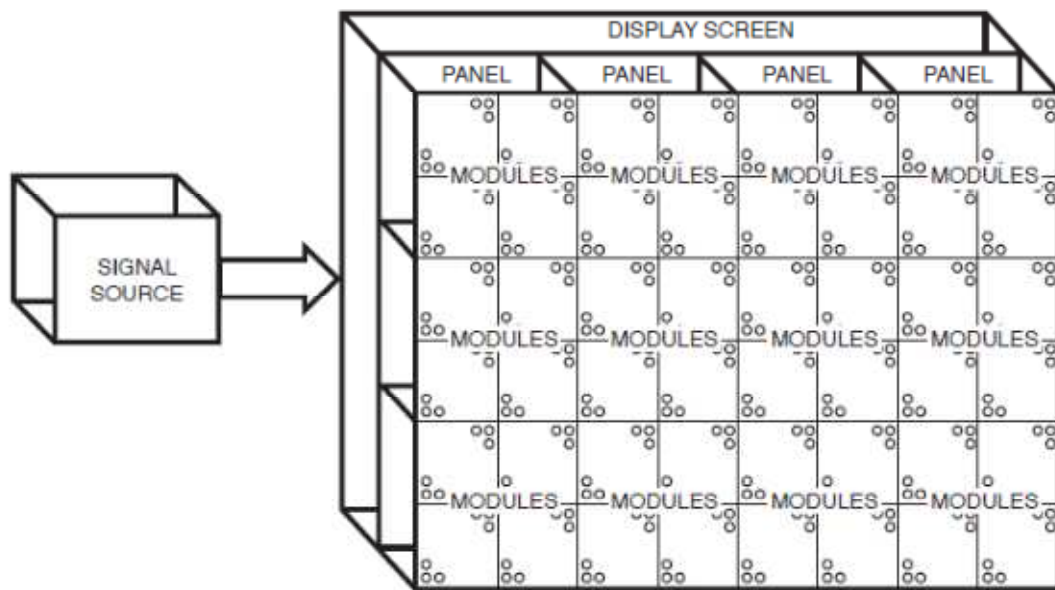


Figura 3.8. Estructura típica de una pantalla LED.

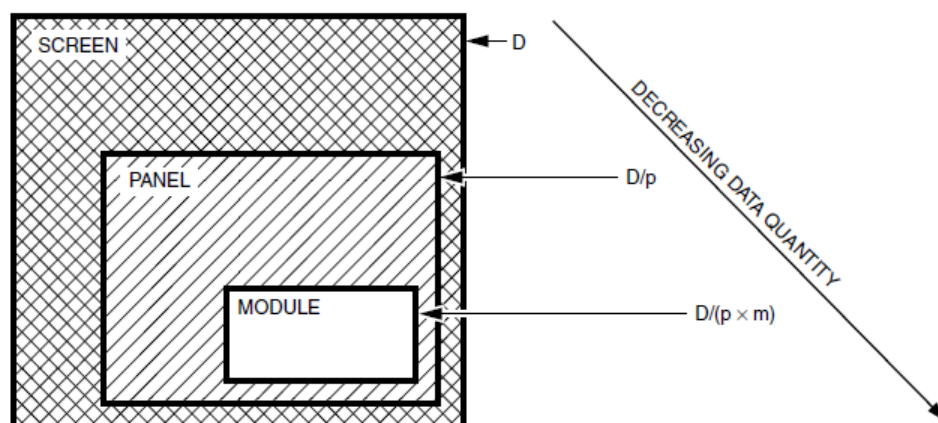


Figura 3.9. Jerarquía de una pantalla. La pantalla consta de varios paneles mientras que un panel consta de varios módulos. D es la cantidad de datos por pantalla, por ejemplo, en una pantalla de 512x384 píxeles con una profundidad de color de 24 bits, $D = 512 \times 384 \times 24 = 4.72$ millones de bits. p = número de paneles por pantalla y m = número de módulos por panel.

3.2.2 El módulo LED

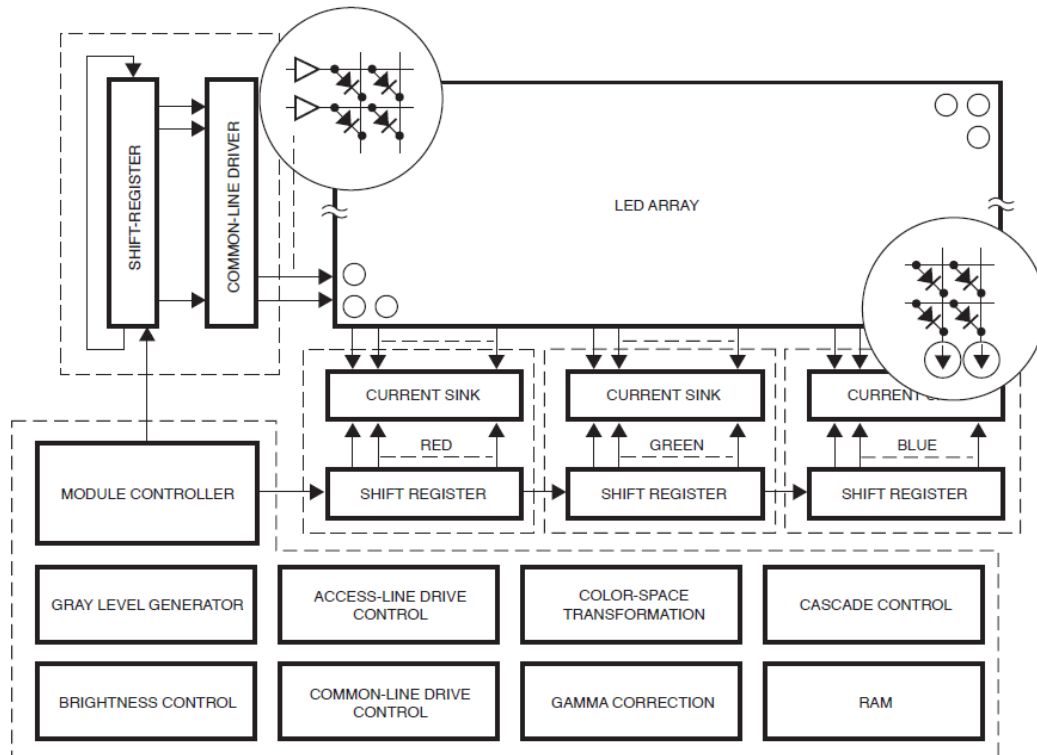


Figura 3.10. Estructura interna de un controlador de módulo.

El controlador del módulo recibe los datos de vídeo y los procesa (por ejemplo corrección gamma). Luego distribuye los datos a los drivers de línea común (lado de arriba) y acceso de línea (lado de abajo). El controlador también es el responsable de producir las señales de control que gobiernan la operación de los drivers. El controlador se puede implementar con una solución FPGA o ASIC.

El driver de línea común actúa como un interruptor de la línea de potencia. Cuando se usa en combinación con el controlador del módulo, el driver de línea común energizará una línea común en un determinado momento (acción de multiplexación) y se sincronizará con los drivers de acceso de línea para asegurar de que se alimentan los datos correctos de la fila. Por lo tanto, los sumideros de corriente adjuntos al acceso de línea solo pueden controlar una fila de LEDs en un determinado momento.

Los drivers de acceso de línea son sumideros de corriente en una matriz de LED de ánodo común. Determinan que LED de la fila de la línea común actualmente energizada debe encenderse. Los drivers incorporan un registro de desplazamiento que permite una interfaz serie. Los sumideros de corriente se preajustan a un valor particular ajustando una resistencia variable externa. Este valor será el valor de la corriente directa de pico del LED. Los sumideros de corriente se activan con el valor de salida del registro de desplazamiento. La generación de la escala de grises involucra múltiples operaciones de latch de los datos de la línea de acceso dentro de un periodo de escaneo.

El controlador del módulo también gestiona otras funciones clave que se describen a continuación:

- **Grayscale control:** La escala de grises son los niveles de brillo que se pueden alcanzar para cada LED en un píxel. Un sistema típico a color puede tener hasta 256 niveles de grises. Esto significa que el brillo de cada LED se puede ajustar de brillo mínimo a máximo en 256 pasos. Si cada píxel tiene tres LED, rojo, verde y azul, el número de combinaciones de colores es $256 \times 256 \times 256 = 16.7$ millones. La escala de grises se logra dividiendo cada período de escaneo en 256 intervalos de tiempo. Por lo tanto, el controlador de acceso de línea se carga con datos 256 veces en un cada periodo de escaneo.
- **Brightness control:** El control de brillo es diferente al de la escala de grises. El control de brillo se refiere al control del valor de luminancia general de la pantalla, no al de un LED individual. Manipulando la duración del período de escaneo se puede controlar la luminancia o brillo general.
- **RAM:** Almacena los datos de video entrantes. Por lo general, se emplea un método de doble buffer porque los datos de video recibidos todavía necesitan ser procesados. Entonces, mientras se usa un buffer para controlar la pantalla, el segundo buffer contiene los datos recibidos recientemente para el procesamiento. Estos dos procesos pueden suceder en paralelo.
- **Gamma correction:** Corrige la función de transferencia no lineal de la pantalla LED. Dicho de otra manera, la transferencia de señal entre los componentes eléctricos y ópticos del sistema de visualización no es lineal. Esto lleva a la expansión de la región brillante y la compresión de la región tenue. Las señales de video NTSC y PAL son corregidas con gamma antes de la transmisión para eliminar el efecto no lineal de la pantalla. Por lo tanto, la pantalla debe tomar esto en cuenta para obtener una función de transferencia de señal lineal. Este tema se analiza con más detalle en la siguiente sección.
- **Color-space transformation:** Es necesario con módulos de colores intensos porque su espacio de color es limitado en comparación con módulos a todo color. Por esta razón, los módulos de colores intensos no se pueden usar para mostrar vídeo a todo color. Se puede usar en videos de color limitado como dibujos animados.
- **Cascade control:** Genera las señales de control utilizadas para interactuar con los otros módulos.

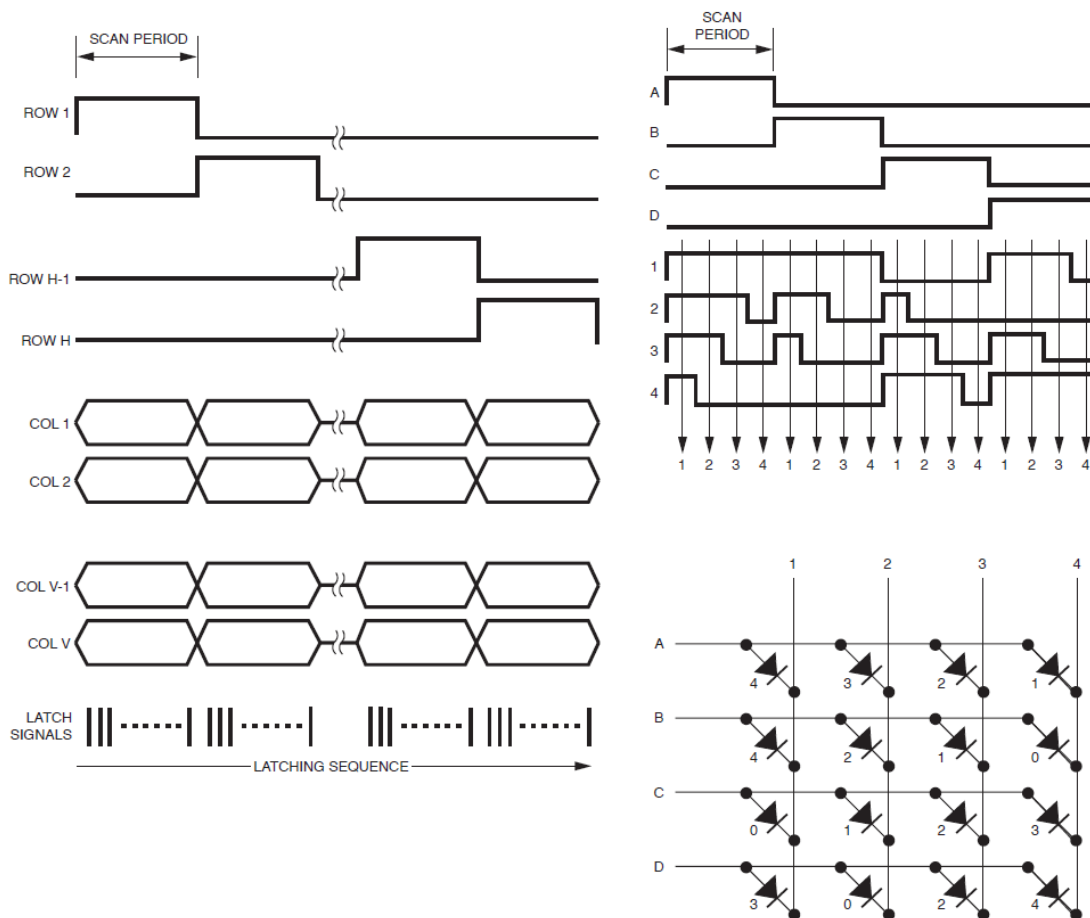


Figura 3.11. Implementación del control de la escala de grises.

Las dos figuras de la derecha muestran la secuencia de tiempo utilizada para producir un sistema de cuatro niveles de grises. Se observa desde la figura superior derecha que los datos se cargan en los controladores de acceso de línea cuatro veces en un período de escaneo único. Eso permite el control del ancho de pulso aplicado a cada LED individualmente. Se producen cuatro niveles de grises. Aunque el ejemplo es de una matriz LED 4x4, el mismo concepto se aplica a una matriz LED más grande.

Por lo tanto, si se requieren más niveles de grises, simplemente hay que aumentar la cantidad de veces que se registran los datos en los drivers de acceso de línea dentro de un solo período de escaneo.

Los datos se registran en los drivers de acceso de línea 256 veces por periodo de escaneo para un sistema con escala de grises de 256 niveles. Esto permite que se pueda controlar el nivel de brillo individual de los LEDs en 256 pasos. Por ejemplo, un LED que está encendido en cada uno de los 256 registros de datos estará con brillo máximo. Si el LED está encendido menos de 256 registros estará tenue.

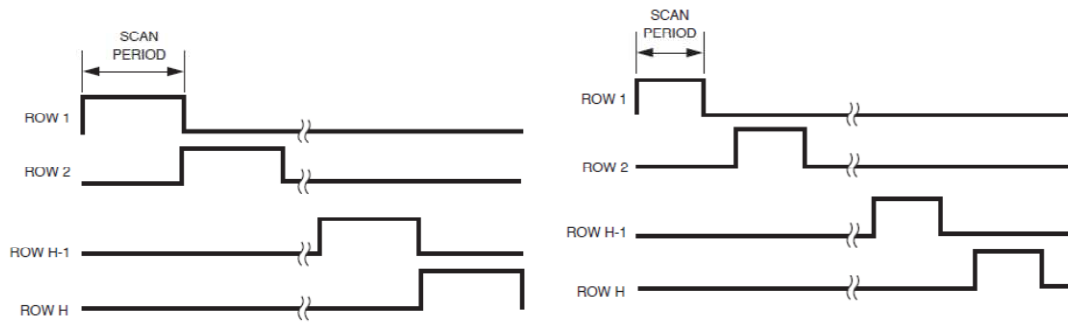


Figura 3.12. Control del brillo general manipulando la longitud del periodo de escaneo.

Una forma de controlar el brillo general es manipulando la longitud del periodo de escaneo. Todos los registros de datos deben estar contenidos en un solo periodo de escaneo. Si el periodo de escaneo se acorta para reducir el brillo general de la pantalla (Figura 3.12), el controlador del módulo debe asegurar de que todos los registros de datos se produzcan dentro de ese periodo de escaneo acortado.

Aunque más caros, los llamados controladores inteligentes reducen la complejidad del controlador del módulo considerablemente. Estos administran la generación de la escala de grises. El controlador del módulo, sin embargo, todavía necesita administrar el control de línea común. El control del brillo también está fuera de los controladores inteligentes. Los controladores inteligentes son muy recomendables para aplicaciones de video.

3.2.2.1 Corrección gamma

La función de transferencia entre los componentes eléctricos y ópticos de un sistema de visualización no es lineal. Si esta no linealidad no se compensa, las regiones brillantes están expandidas y las oscuras están comprimidas.

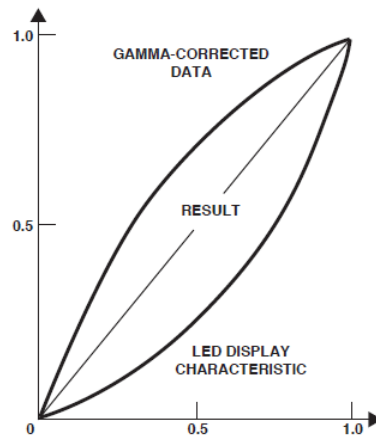


Figura 3.13. Corrección Gamma.

Como ejemplo, las pantallas CRT tienen un gamma de alrededor de 0.45. Por lo tanto:

$$R_{\text{display}} = R_{\text{in}}^{2.2}$$

$$G_{\text{display}} = G_{\text{in}}^{2.2}$$

$$B_{\text{display}} = B_{\text{in}}^{2.2}$$

donde R_{in} , G_{in} y B_{in} son las entradas de datos a los controladores de la pantalla. Si hacemos correcciones gamma para R_{in} , G_{in} y B_{in} :

$$R_{\text{in}}' = R_{\text{in}}^{0.45}$$

$$G_{\text{in}}' = G_{\text{in}}^{0.45}$$

$$B_{\text{in}}' = B_{\text{in}}^{0.45}$$

donde R_{in}' , G_{in}' y B_{in}' son las señales de video entrantes desde la fuente de video:

$$R_{\text{display}} = (R_{\text{in}}')^{2.2} = R_{\text{in}}'$$

$$G_{\text{display}} = (G_{\text{in}}')^{2.2} = G_{\text{in}}'$$

$$B_{\text{display}} = (B_{\text{in}}')^{2.2} = B_{\text{in}}'$$

Entonces se establecerá una relación lineal. La corrección gamma usualmente se implementa usando una tabla de búsqueda, por ejemplo, PROM o DSP. La calidad de la imagen es una medida subjetiva. Los fabricantes de monitores suelen ajustar sus pantallas para un brillo, contraste y gamma aceptables.

3.2.2.2 Transformación del espacio de color

Las pantallas de colores intensos poseen un espacio de color significativamente más pequeño que las pantallas a todo color. Si las señales de video entrantes suponen que la pantalla que están controlando tiene un espacio a todo color, los colores en la pantalla de colores intensos aparecen comprimidos. Se puede implementar una corrección de color usando también una tabla de búsqueda.

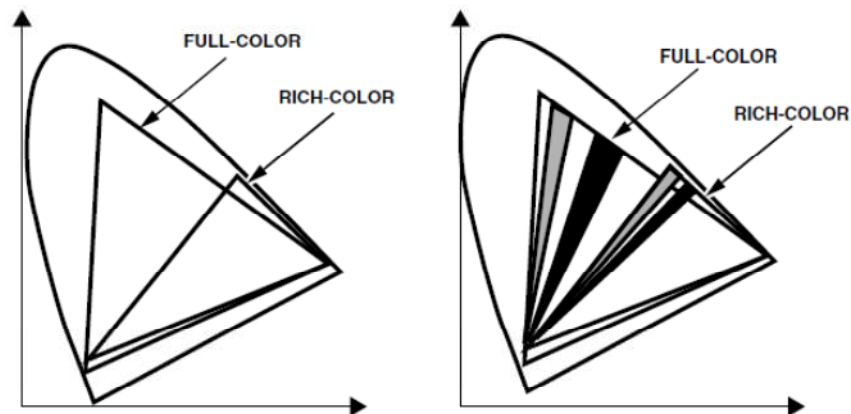


Figura 3.14. Diagramas cromáticos CIE.

3.2.2.3 Distribución de datos

Como se mencionó anteriormente, es mejor descomponer un sistema de pantalla en el nivel de módulo y en el nivel de panel. Cada uno es responsable de la distribución de datos en su propio nivel. El controlador del módulo es el responsable de distribuir los datos dentro de su panel y un controlador de panel dado sólo recibe datos relevantes para ese panel dado. El controlador debería ignorar todos los demás datos. Esto simplifica la gestión de datos. Un posible método incluiría un modelo basado en Ethernet y un bus de datos basado en línea de direcciones.

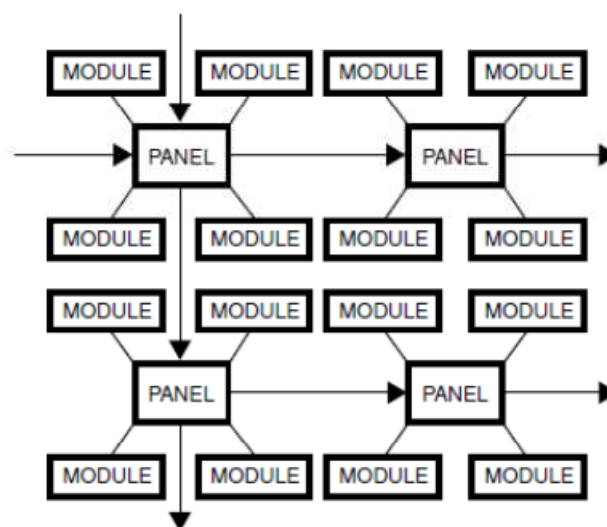


Figura 3.15. Distribución de datos.

4 CARACTERIZACIÓN Y FUNCIONAMIENTO DE UN MÓDULO LED

4.1 Descripción

Se va a describir un módulo LED de 64x64 píxeles (Pixel Pitch 3.91) y escaneo de 1/16 como el mostrado anteriormente y el mismo que se utilizará en el diseño del controlador.

El módulo consta de 4096 LEDs RGB distribuidos en 4 conjuntos de 1024 LEDs.

Cada conjunto de 1024 LEDs está compuesto a su vez de 16 filas de 64 LEDs cada una, dispuestos según una configuración matricial con fila de ánodo común.

Las filas 1 a 16 de cada matriz se barren a la vez (escaneo de 1/16), por lo que en el módulo se actualizan 4 filas a la vez.

Para el control del módulo se necesitan 19 señales:

- 3 señales RGB (grupo de datos) para cada matriz, es decir, 12 señales en total para llevar los datos de imagen a los drivers de acceso de línea.
- 3 señales (CLK, LATCH y OE) para el control de los drivers de acceso de línea.
- 4 señales (A, B, C y D) para la multiplexación de las filas.

Para la alimentación del módulo se emplea una fuente de 4,2V y 15A.

4.2 Funcionamiento

Todas las señales se distribuyen a través de transceptores de bus octales de alta corriente 74HC245, con el objetivo de acondicionarlas y potenciarlas, minimizando los requerimientos externos de temporización.

Las señales A, B, C y D llegan a dos demultiplexores 74HC138 de 3 entradas y 8 salidas. La señal D controla la habilitación de los demultiplexores, permitiendo así la operación en cascada y la obtención de un demultiplexor con 4 entradas y las 16 salidas necesarias para la selección de las filas activas.

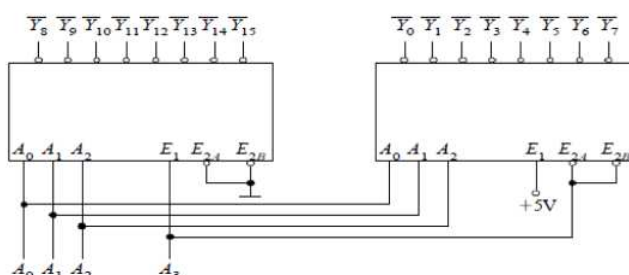


Figura 4.1. Demultiplexores en cascada.

Las salidas de los demultiplexores gobiernan las puertas de los MOSFET MBI5907, que son los encargados de la conmutación de potencia de las filas. El MBI5907 es un conmutador de potencia específico para el escaneo en pantallas de LED. Tiene 2 canales de salida, opera desde 3.3V a 5V y soporta un máximo de 2A por canal.

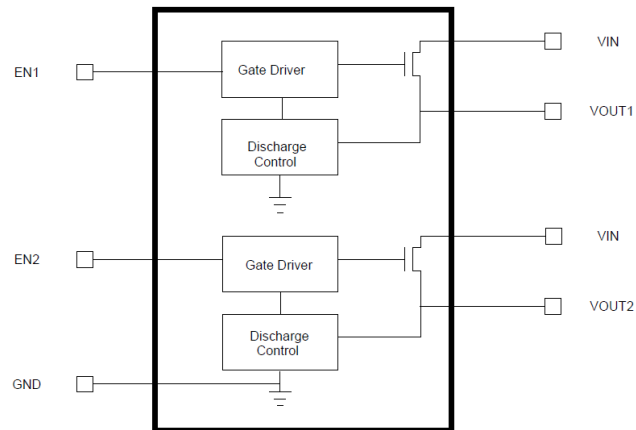


Figura 4.2. Diagrama de bloques del MOSFET MBI5907.

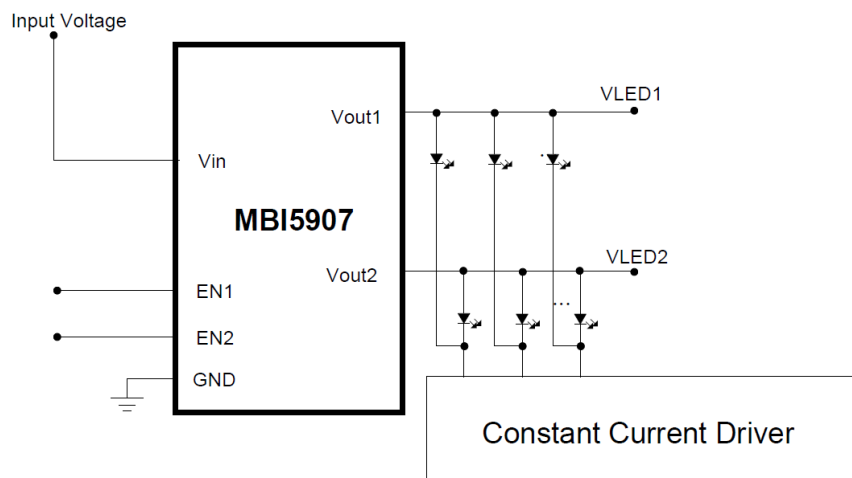


Figura 4.3. Circuito de aplicación típico del MOSFET MBI5907.

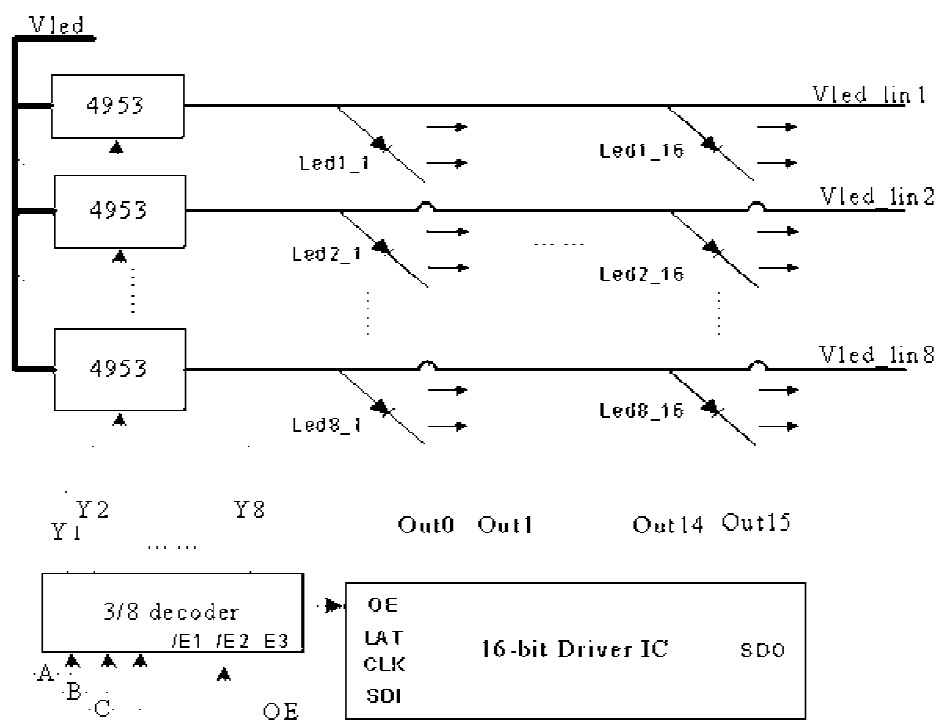


Figura 4.4. Principio de escaneo dinámico (multiplexación en tiempo) que permite que un canal del driver pueda controlar más de un LED.

Las señales R, G y B, en cada grupo de datos, llegan hasta 3 conjuntos (uno para componente de color) de 4 drivers MBI5024 en cascada. Cada driver consta de 16 salidas, obteniéndose así las 64 salidas necesarias para controlar las 64 columnas de la matriz de LEDs. La corriente de salida de los drivers se ajusta a través de una resistencia conectada a la patilla R-EXT.

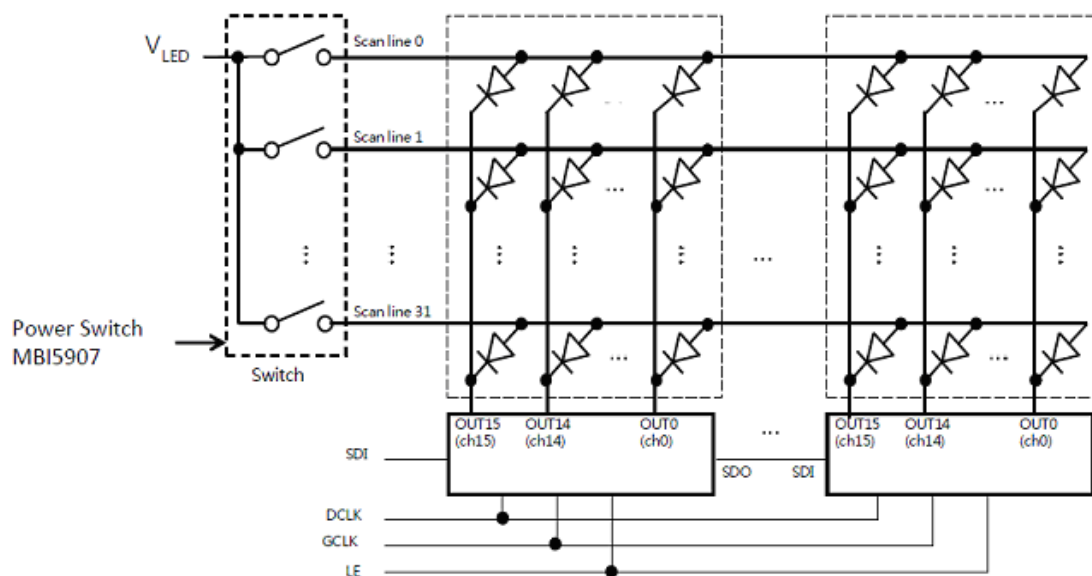


Figura 4.5. Drivers MBI5024 en cascada.

Elegir un buen driver es el primer paso para conseguir una buena pantalla LED de alta definición. El driver representa menos del 5% del BOM (Bill of Materials), pero influye mucho en el rendimiento. En nuestro caso, el driver MBI5024 está diseñado para pantallas LED que requieren operación a baja corriente e igualdad en la intensidad luminosa de cada canal. Sus principales características son:

- 16 canales de salida de corriente constante e invariante con los cambios de voltaje de la carga, con rango de:
 - 3-45mA @ VDD = 5V.
 - 3-30mA @ VDD = 3.3V.
- Excelente precisión de corriente de salida:
 - entre canales: $\pm 1.5\%$ (típico) y $\pm 2.5\%$ (máximo).
 - entre circuitos integrados: $\pm 1.5\%$ (típico) y $\pm 3\%$ (máximo).
- Corriente de salida ajustada a través de una resistencia externa (R-EXT).
- Respuesta rápida de la corriente de salida, OE (min.): 70ns con buena uniformidad entre los canales de salida.
- Demora escalonada de salida.
- Frecuencia de reloj de 25MHz.
- Entrada con disparo Schmitt.

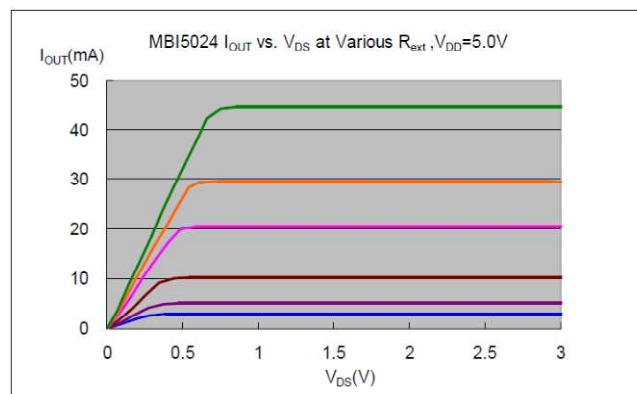


Figura 4.6. Corriente constante en driver MBI5024.

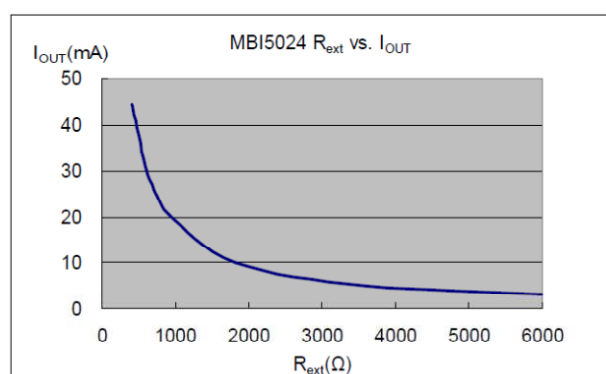


Figura 4.7. Corriente de salida ajustable en driver MBI5024.

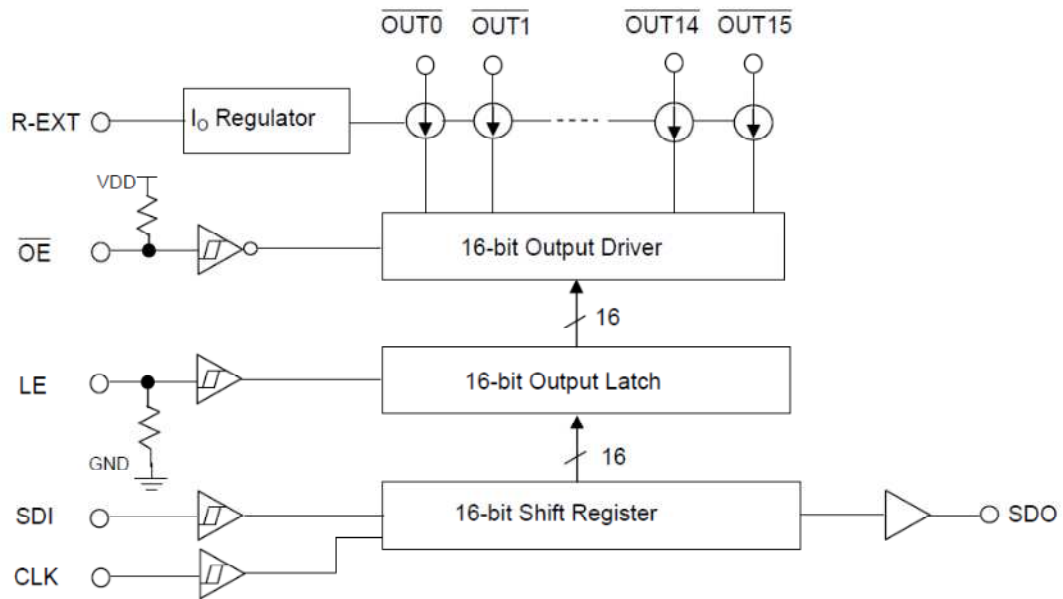


Figura 4.8. Diagrama de bloques de driver MBI5024.

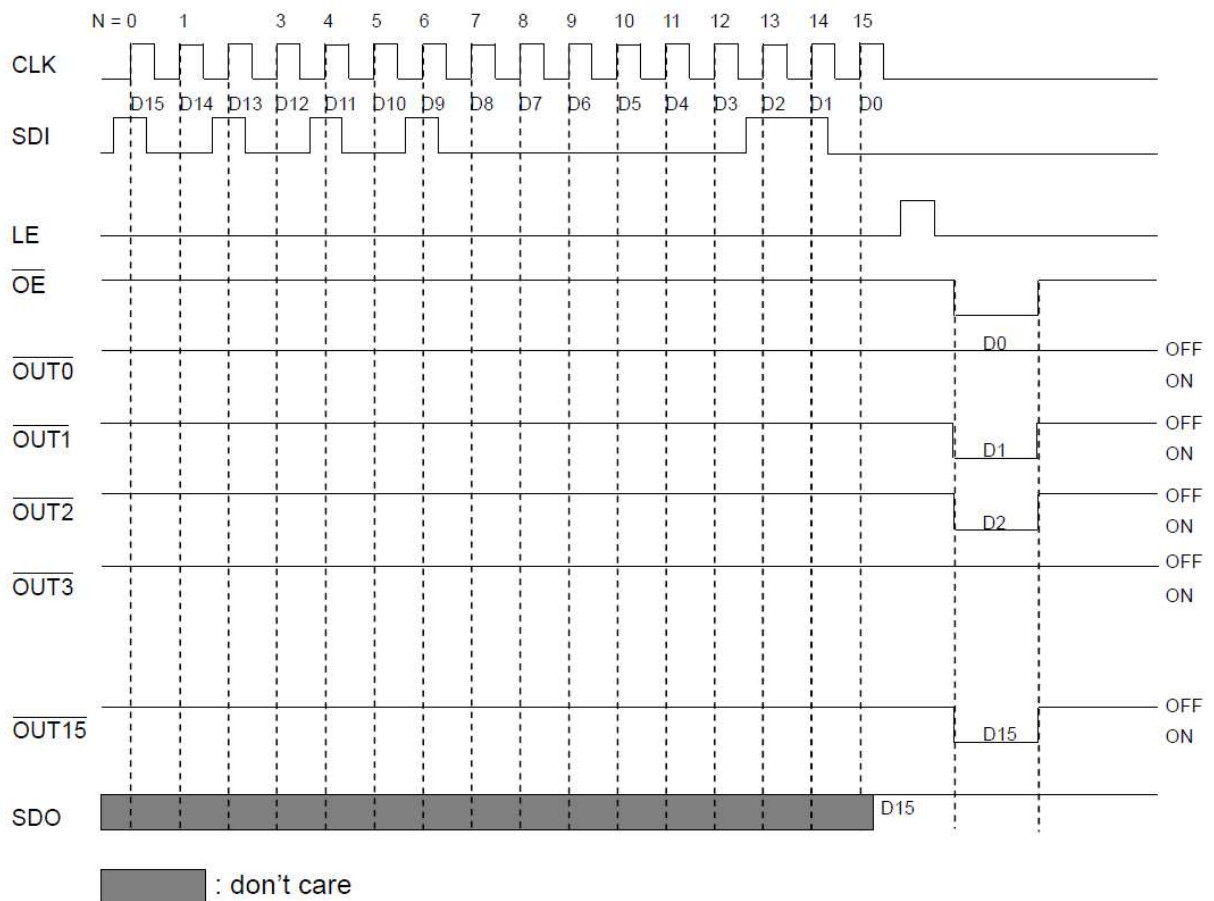


Figura 4.9. Diagrama de tiempos de driver MBI5024.

CLK es la señal de reloj que sincroniza la operación de los drivers y suministra los pulsos de cambio para los registros de desplazamiento. Por este motivo, la frecuencia de las señales de datos ha de ser la mitad de la frecuencia de la señal de reloj.

La señal de LATCH es la encargada de capturar la información para ser representada en la fila, lo cual ocurre cada vez que se llena el registro de desplazamiento formado por la cascada de 4 drivers. Para una escala de grises de 256 niveles, la señal de LATCH se accionará 256 veces durante la actualización de la fila, para accionar los LEDs con la señal del PWM con la que se consigue la regulación de la luminosidad.

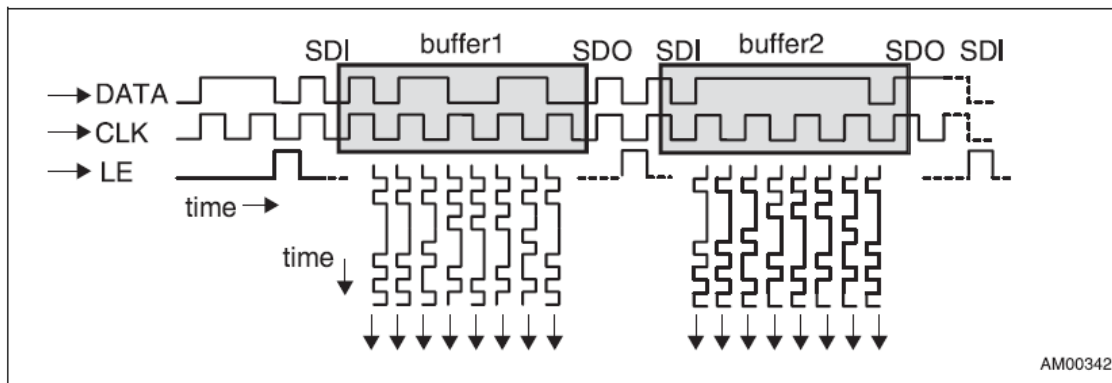


Figura 4.10. Señales de control en drivers.

Por último, la señal OE desactiva los drivers para realizar la conmutación de fila.

5 IMPLEMENTACIÓN DE UN CONTROLADOR PARA MÓDULO DE PANTALLA LED BASADO EN FPGA

5.1 Introducción

En este capítulo se describen los pasos seguidos en el diseño y desarrollo para implementar el control de un módulo de pantalla LED como el descrito en el capítulo anterior.

La elección de una FPGA como el núcleo del sistema, que hace que el diseño sea más complejo de realizar, se debe a la falta de flexibilidad de un microprocesador que obligaría a alterar drásticamente el diseño del sistema cuando se cambia el tamaño de la pantalla LED.

Aunque sólo se ha implementado el control para un módulo, debido al elevado rendimiento y flexibilidad de la FPGA, la extensión del control para un panel completo (4 módulos) y una pantalla completa (por ejemplo 16x9 paneles) sólo supone unas modificaciones del diseño fáciles y rápidas.

5.2 Objetivos de diseño

Como objetivos de diseño del controlador se tienen los siguientes puntos:

- Como fuente de información se desarrollará una aplicación para PC que permita el envío de patrones de prueba, imágenes y vídeo de baja velocidad.
- El envío de la información desde el PC se realizará a través de una interfaz Ethernet.
- El controlador deberá disponer de una función para la corrección Gamma.
- Se deberá alcanzar una escala de grises de 256 niveles.
- La pantalla deberá actualizarse 60 veces por segundo.

5.3 Hardware del sistema

5.3.1 Módulo de evaluación para FPGA

Para la implementación del controlador se emplea un módulo de evaluación para FPGA Spartan-3E, disponible en el departamento de ingeniería electrónica de la escuela, como el que se muestra a continuación.



Figura 5.1. Módulo de evaluación para FPGA Spartan-3E.

Además de la FPGA se utilizan los siguientes recursos incluidos en el módulo de evaluación:

Reloj del sistema

Consiste en un oscilador de 50 MHz (Figura 5.2) con un ciclo de trabajo de entre el 40% y el 60% y una precisión de ± 2500 Hz o ± 50 ppm. La entrada del reloj está conectada directamente a una entrada de buffer global y a un DCM (Digital Clock Manager) asociado.



Figura 5.2. Oscilador de 50 MHz incluido.

Interfaz de Capa Física Ethernet 10/100

El módulo de evaluación también incluye un interfaz de capa física Ethernet 10/100 (PHY) LAN83C185 de Standard Microsystems y un conector RJ-45, como se muestra en la Figura 5.3. Con un controlador de acceso al medio Ethernet (MAC) implementado en la FPGA, el módulo se puede conectar a una red Ethernet estándar. Toda la temporización se controla desde un oscilador de cristal de 25 MHz también incluido en el módulo de evaluación.



Figura 5.3. 10/100 Ethernet PHY con conector RJ-45 y oscilador de cristal de 25MHz.

La FPGA se conecta al Ethernet PHY LAN83C185 mediante MII (Media Independent Interface) estándar, como se muestra en la Figura 5.4 y en la Tabla 5.1 con el detalle de las señales, incluido el número de patilla de la FPGA.

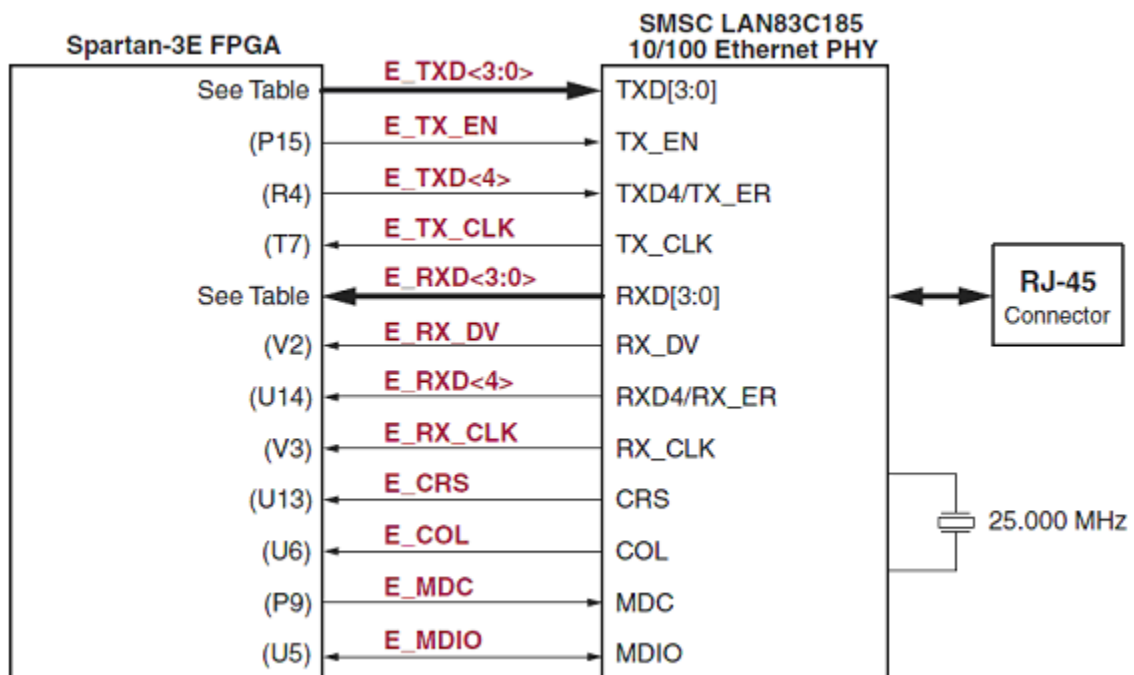


Figura 5.4. Conexión de la FPGA al Ethernet PHY vía MII.

Tabla 5.1. Conexiones de la FPGA al Ethernet PHY LAN83C185.

Signal Name	FPGA Pin Number	Function
E_TXD<4>	R6	Transmit Data to the PHY. E_TXD<4> is also the MII Transmit Error.
E_TXD<3>	T5	
E_TXD<2>	R5	
E_TXD<1>	T15	
E_TXD<0>	R11	
E_TX_EN	P15	Transmit Enable.
E_TX_CLK	T7	Transmit Clock. 25 MHz in 100Base-TX mode, and 2.5 MHz in 10Base-T mode.
E_RXD<4>	U14	Receive Data from PHY.
E_RXD<3>	V14	
E_RXD<2>	U11	
E_RXD<1>	T11	
E_RXD<0>	V8	
E_RX_DV	V2	Receive Data Valid.
E_RX_CLK	V3	Receive Clock. 25 MHz in 100Base-TX mode, and 2.5 MHz in 10Base-T mode.
E_CRS	U13	Carrier Sense
E_COL	U6	MDIO Collision Detect.
E_MDC	P9	Management Clock. Serial management clock.
E_MDIO	U5	Management Data Input/Output.

Conectores de expansión y LEDs

El módulo de evaluación dispone de varios conectores de expansión para flexibilizar una interfaz sencilla a otros componentes externos, como 3 conexiones de 6 patillas para módulos periféricos, un conector Hirose FX2 con 43 patillas de entrada/salida de usuario asociadas, así como puertos VGA, PS2 y RS232. El módulo de evaluación también incluye 8 LEDs individuales de montaje superficial.

**Figura 5.5. Conectores de expansión y LEDs.**

5.3.2 Diseño y desarrollo del hardware del controlador

En la Figura 5.6 se muestra una representación del bloque del controlador que se ha diseñado, donde se pueden ver las señales de entrada (izquierda) y salida (derecha) las cuales se listan y comentan a continuación:

Entradas:

- clk: Reloj del sistema.
- rx_clk, rx_dv, rx_data: Reloj, dato válido y dato recibido entregados por el interfaz Ethernet PHY LAN83C185 del módulo de evaluación.

Salidas:

- led: LEDs para la representación visual del estado del reset del sistema y del enclavamiento del PLL.
- display_addr: Señales para la multiplexación de las filas del módulo LED.
- display_rgb1, display_rgb2, display_rgb3 y display_rgb4: Señales para los datos de imagen de los 4 grupos de datos.
- display_clk, display_lat, display_oe: Señales de control.

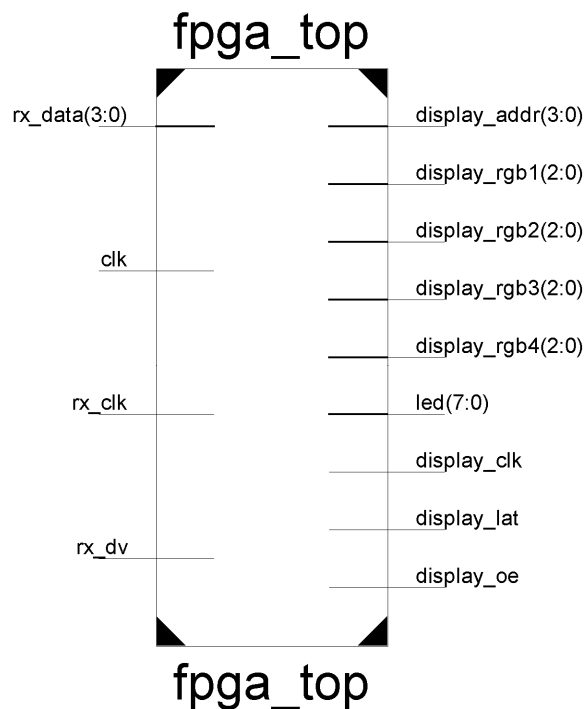


Figura 5.6. Representación del bloque del controlador.

En la Figura 5.7 se muestra el detalle de los módulos que incluye el controlador.

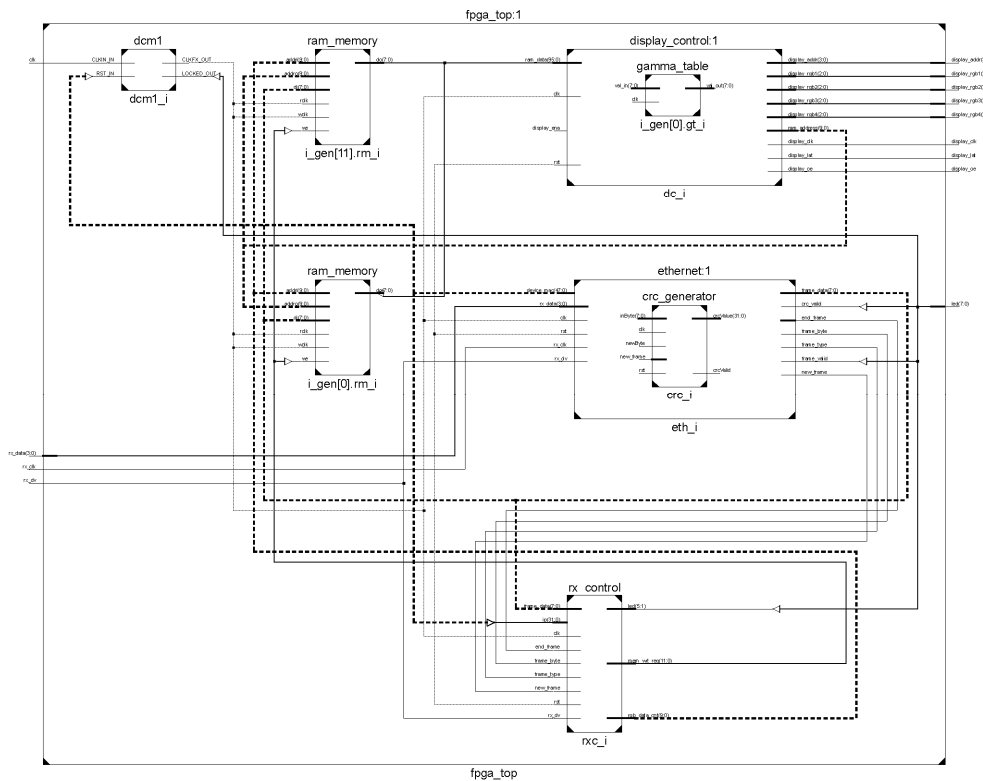


Figura 5.7. Detalle de módulos del controlador.

A continuación se detallan cada uno de los módulos del controlador.

5.3.2.1 Módulo 'ethernet'

El módulo 'ethernet' recoge los datos entregados por el interfaz Ethernet PHY (nivel físico) y entrega los datos a la siguiente capa de la pila de protocolos (nivel de red).

El módulo 'ethernet' se implementa gracias al código 'ethernet.vhd' de los autores **'Ashley Partis and Jorgen Peddersen'**. Este código se encuentra de forma libre en el foro de discusión de la comunidad de usuarios de FPGA de Xilinx.

Recibe las tramas de la interfaz Ethernet PHY y las pasa a la siguiente capa utilizando una secuencia de bytes. Solo permite tramas IP y ARP válidas. Las tramas con CRC incorrectos se declaran inválidas. Las tramas con MAC que no son de difusión o no se corresponden con la definida como constante global se ignoran. La señal rx_err también se ignora.

En la Figura 5.8 se muestra una representación del bloque del módulo 'ethernet', donde se pueden ver las señales de entrada (izquierda) y salida (derecha) las cuales se listan a continuación tal y como están definidas por los autores:

Entradas:

- clk: Clock.
- rst: Asynchornous active low reset.
- rx_clk: PHY receiver clock.
- rx_dv: PHY receive data valid signal.
- rx_data: 4 bit receive data line from the ethernet PHY.
- rx_err: PHY receive error signal.
- device_mac: FPGA board MAC.

Salidas:

- new_frame: New frame signal to the next layer.
- frame_type: Inform the ARP or IP (1) processes when we have a byte.
- frame_byte: Signal to write data to IP layer.
- frame_data: Data to write to the IP layer.
- end_frame: End of frame signal to the next layer.
- frame_valid: Indicate if the frame is valid (assert with end_frame).
- crc_valid: CRC valid.

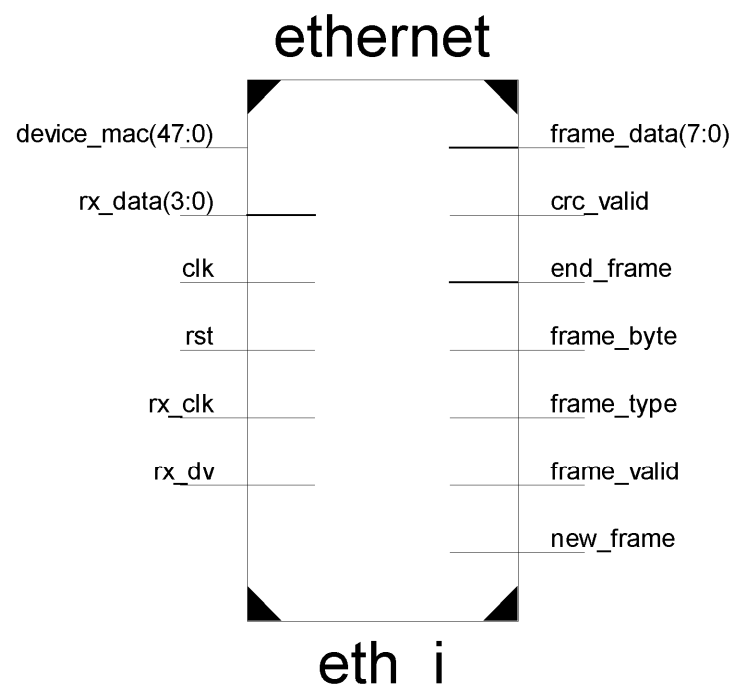


Figura 5.8. Representación del módulo ‘ethernet’.

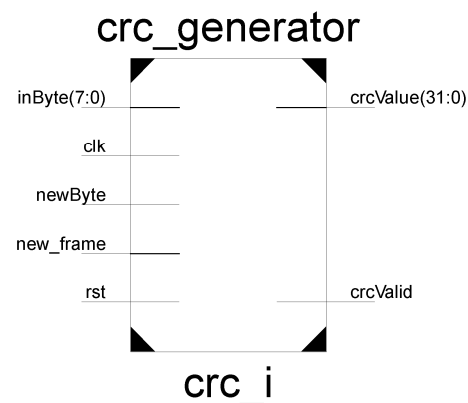
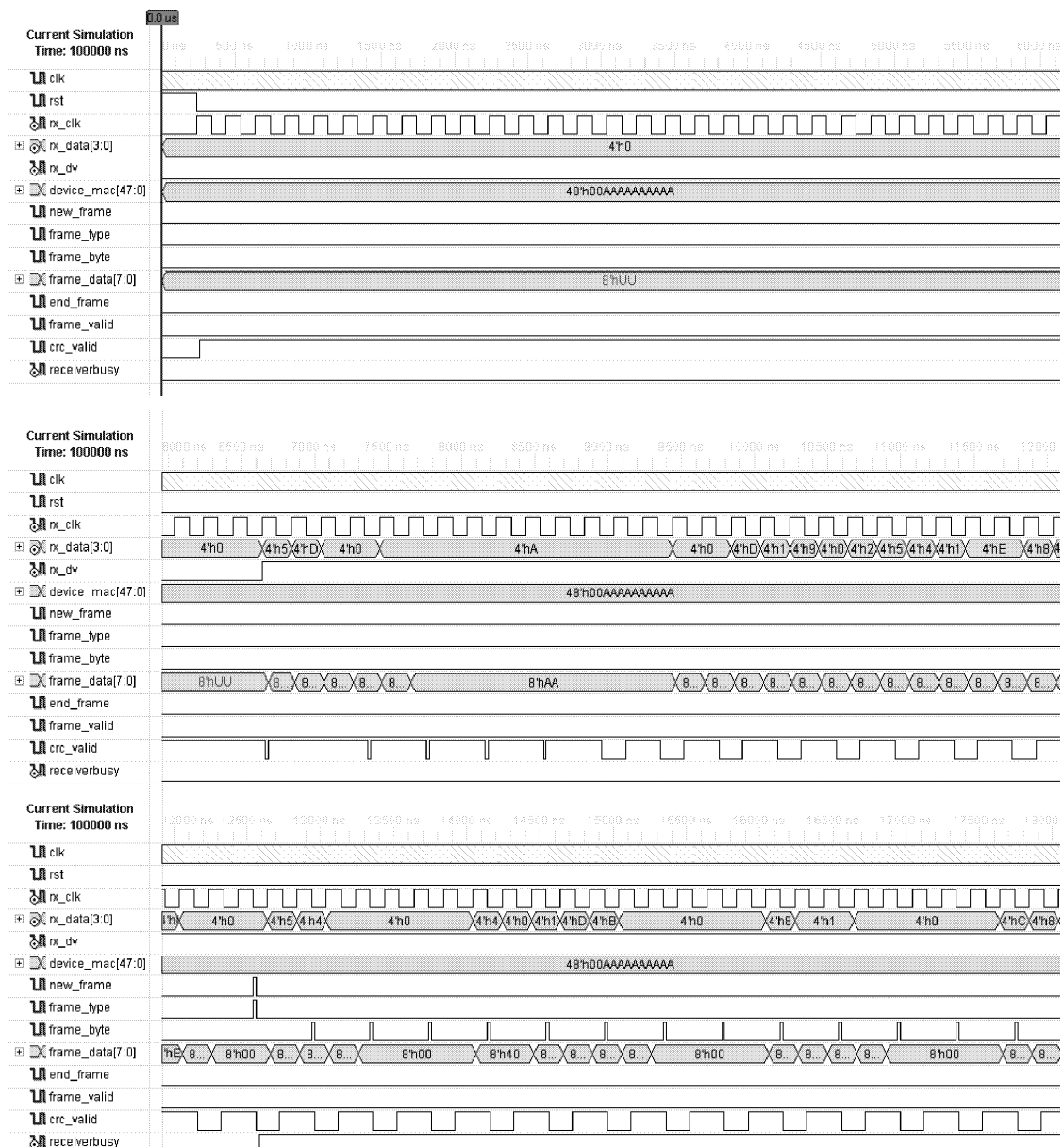
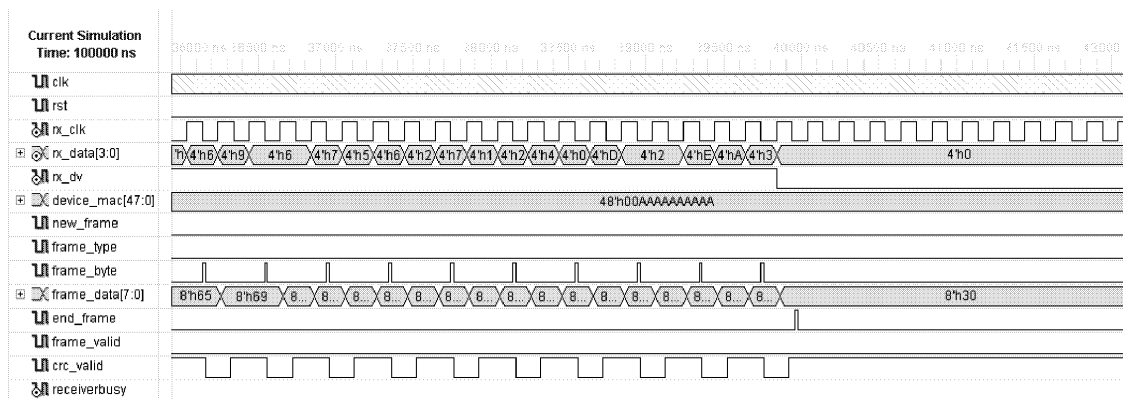
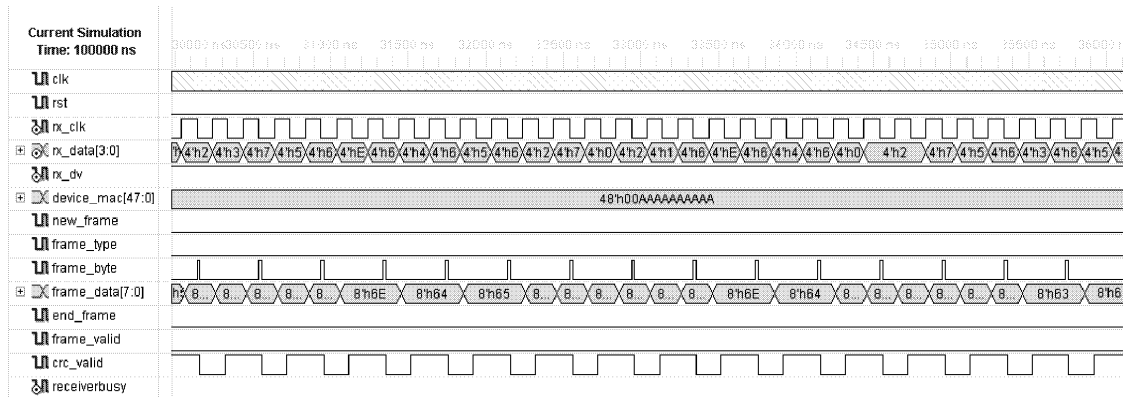
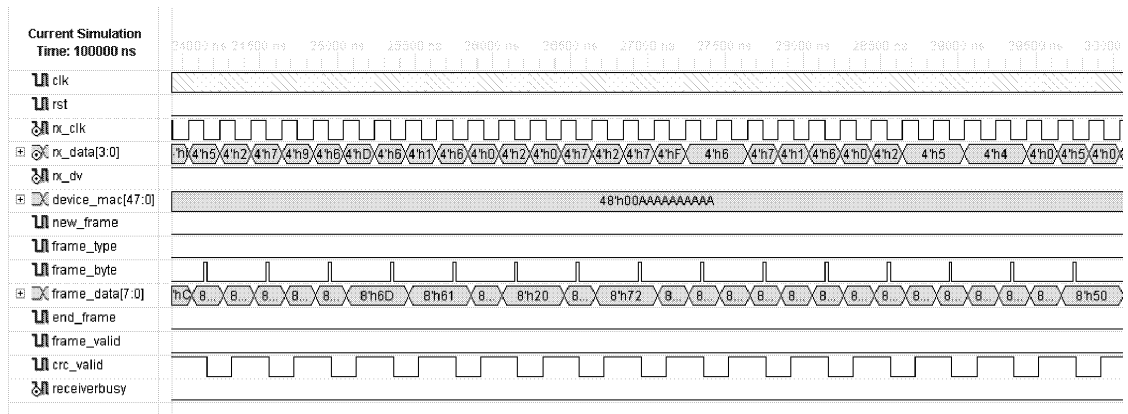
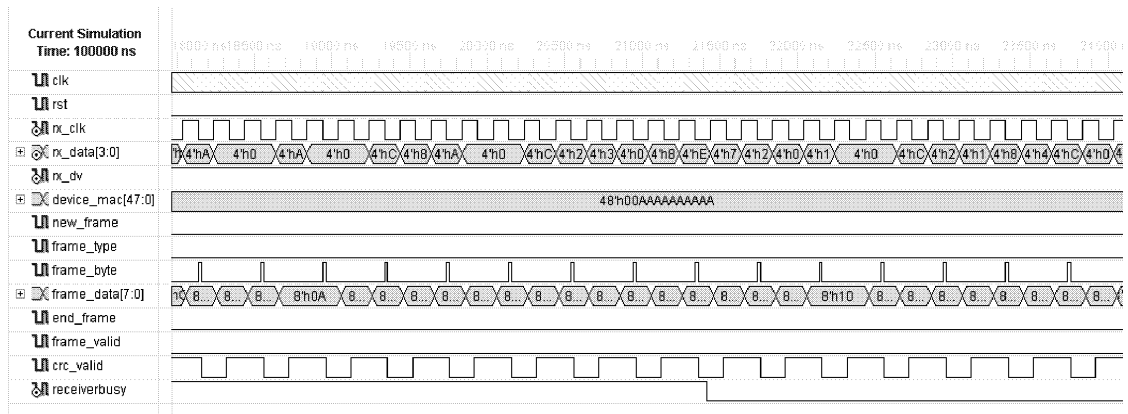


Figura 5.9. Detalle del módulo ‘crc_generator’ incluido en el módulo ‘ethernet’.

A continuación se muestran los resultados de las simulaciones que demuestran el funcionamiento del módulo ‘ethernet’.





5.3.2.2 Módulo 'rx_control'

El módulo 'rx_control' recibe los datos del módulo 'ethernet', analiza el protocolo, extrae los datos y los almacena en una memoria RAM (módulo 'ram_memory') desde la cual se alimenta al módulo de control del escaneo (módulo 'display_control').

En la Figura 5.10 se muestra una representación del bloque del módulo 'rx_control', donde se pueden ver las señales de entrada (izquierda) y salida (derecha) las cuales se listan y comentan a continuación (sólo se muestran las señales que no se han definido en módulos anteriormente descritos):

Entradas:

- ip: Dirección IP del controlador.

Salidas:

- mem_wrt_req: Señales de habilitación para la escritura en el correspondiente bloque de memoria.
- rgb_data_cnt: Contador para el direccionamiento de la escritura en memoria.

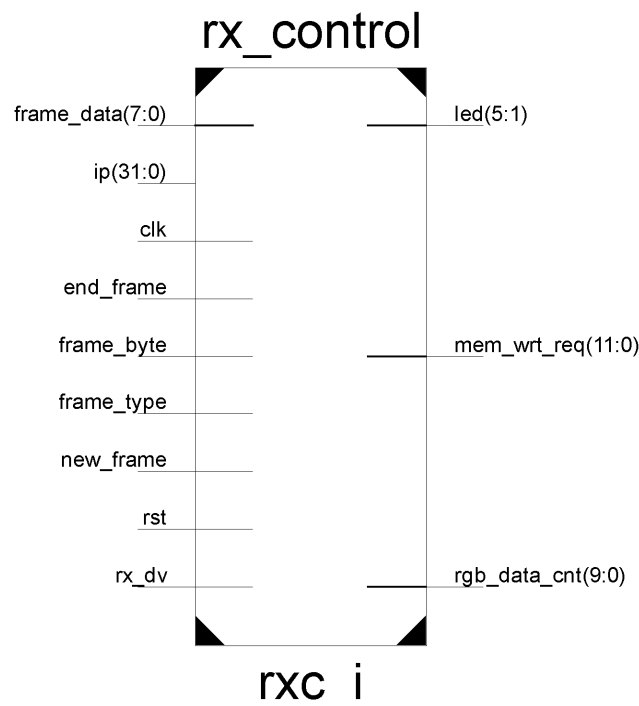


Figura 5.10. Representación del módulo 'rx_control'.

Para implementar el módulo 'rx_control' partimos recordando la Pila de Protocolos TCP/IP hasta el Nivel de Transporte, donde se encuentra el protocolo UDP que se ha empleado para la comunicación entre el PC y la FPGA.



Figura 5.11. Pila TCP/IP hasta el Nivel de Transporte.

El módulo 'ethernet' proporciona la trama Ethernet (Figura 5.12) que encapsula a los protocolos IP y TCP en su Payload (Figura 5.13).

Preámbulo	Delimitador de inicio de trama	MAC de destino	MAC de origen	802.1Q Etiqueta(opcional)	Ethertype (Ethernet II) o longitud (IEEE 802.3)	Payload	Secuencia de comprobación (32-bit CRC)	Gap entre frames
7 Bytes	1 Byte	6 Bytes	6 Bytes	(4 Bytes)	2 Bytes	De 46 (o 42) hasta 1500 Bytes	4 Bytes	12 Bytes
64-1522 Bytes								
72-1530 Bytes								
64-1542 Bytes								

Figura 5.12. Estructura de la trama Ethernet 802.3.

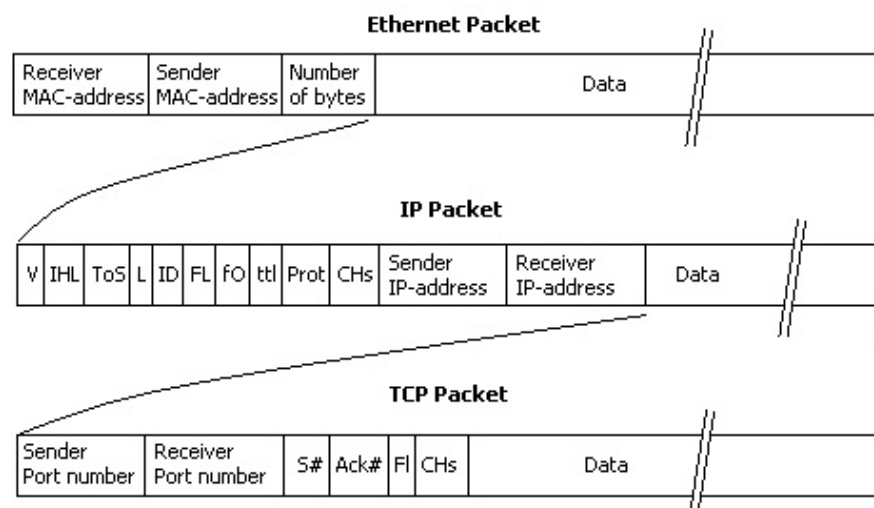


Figura 5.13. Estructura de la Payload en Ethernet y protocolos IP y TCP.

El módulo 'rx_control' identifica cada nueva trama y si se trata de un paquete IP (x"45" - Versión y Tamaño cabecera) extrae su Payload.

0-3	4-7	8-15	16-18	19-31
Versión	Tamaño Cabecera	Tipo de Servicio	Longitud Total	
Identificador			Flags	Posición de Fragmento
Tiempo de Vida		Protocolo	Suma de Control de Cabecera	
Dirección IP de Origen				
Dirección IP de Destino				
Opciones				Relleno

Figura 5.14. Formato de la Cabecera IP (Versión 4).

En el análisis de los datos de la Payload del paquete IP se identifica el Protocolo UDP (x"11") y a continuación la dirección IP, y si hay coincidencia se extraen los datos. Para poder realizar estas dos operaciones a continuación se comentan brevemente los detalles más importantes del protocolo UDP y su cabecera.

UDP (User Datagram Protocol) es un protocolo mínimo de nivel de transporte orientado a mensajes (RFC 768 de la IETF). Permite el envío de datagramas a través de la red sin que se haya establecido previamente una conexión, ya que el propio datagrama incorpora suficiente información de direccionamiento en su cabecera. Tampoco tiene confirmación ni control de flujo, por lo que los paquetes pueden adelantarse unos a otros; y tampoco se sabe si han llegado correctamente, ya que no hay confirmación de entrega o recepción.

UDP proporciona una sencilla interfaz entre la capa de red y la capa de aplicación. No otorga garantías para la entrega de sus mensajes y el origen no retiene estados de los mensajes que han sido enviados a la red. Cualquier tipo de garantías para la transmisión de la información deben ser implementadas en capas superiores. Su uso principal es para la transmisión de audio y vídeo en tiempo real, donde no es posible realizar retransmisiones por los estrictos requisitos de retardo que se tiene en estos casos. Resulta más importante transmitir con velocidad que garantizar el hecho de que lleguen absolutamente todos los bytes.

La cabecera UDP consta de 4 campos de los cuales 2 son opcionales (sombreados en la Figura 5.15). Los campos de los puertos origen y destino son campos de 16 bits que identifican el proceso de emisión y recepción. Ya que UDP carece de un servidor de estado y el origen UDP no solicita respuestas, el puerto origen es opcional. En caso de no ser utilizado, el puerto origen debe ser puesto a cero. A los campos del puerto destino le sigue un campo obligatorio que indica el tamaño en bytes del datagrama UDP incluidos los datos. El valor mínimo es de 8 bytes. El campo de la cabecera restante es una suma de comprobación de 16 bits que abarca una pseudo-cabecera IP (Figura 5.16, con las IP origen y destino, el protocolo y la longitud del paquete UDP), la cabecera UDP, los datos y 0's hasta completar un múltiplo de 16. El checksum también es opcional en IPv4, aunque generalmente se utiliza en la práctica (en IPv6 su uso es obligatorio). A continuación se muestran los campos para el cálculo del checksum en IPv4, sombreada la pseudo-cabecera IP.

+	Bits 0 - 15	16 - 31
0	Puerto origen	Puerto destino
32	Longitud del Mensaje	Suma de verificación
64	Datos	

Figura 5.15. Cabecera UDP.

bits	0 – 7	8 – 15	16 – 23	24 – 31
0	Dirección Origen			
32	Dirección Destino			
64	Ceros	Protocolo	Longitud UDP	
96	Puerto Origen		Puerto Destino	
128	Longitud del Mensaje		Suma de verificación	
160	Datos			

Figura 5.16. Pseudo-Cabecera IP.

Por último, como forma de garantizar que se reciben datos validos para nuestra aplicación, se ha implementado un protocolo sencillo consistente en una cabecera de dos Bytes (x"AAAA") que identifica los datos de vídeo y un tercer Byte para la selección de la memoria (componente de color y grupo de datos) que se va a actualizar.

En los siguientes diagramas de flujo se muestra el funcionamiento de la máquina de estados con la que se implementa el módulo 'rx_control'. Para mayor claridad se ha dividido en tres partes.

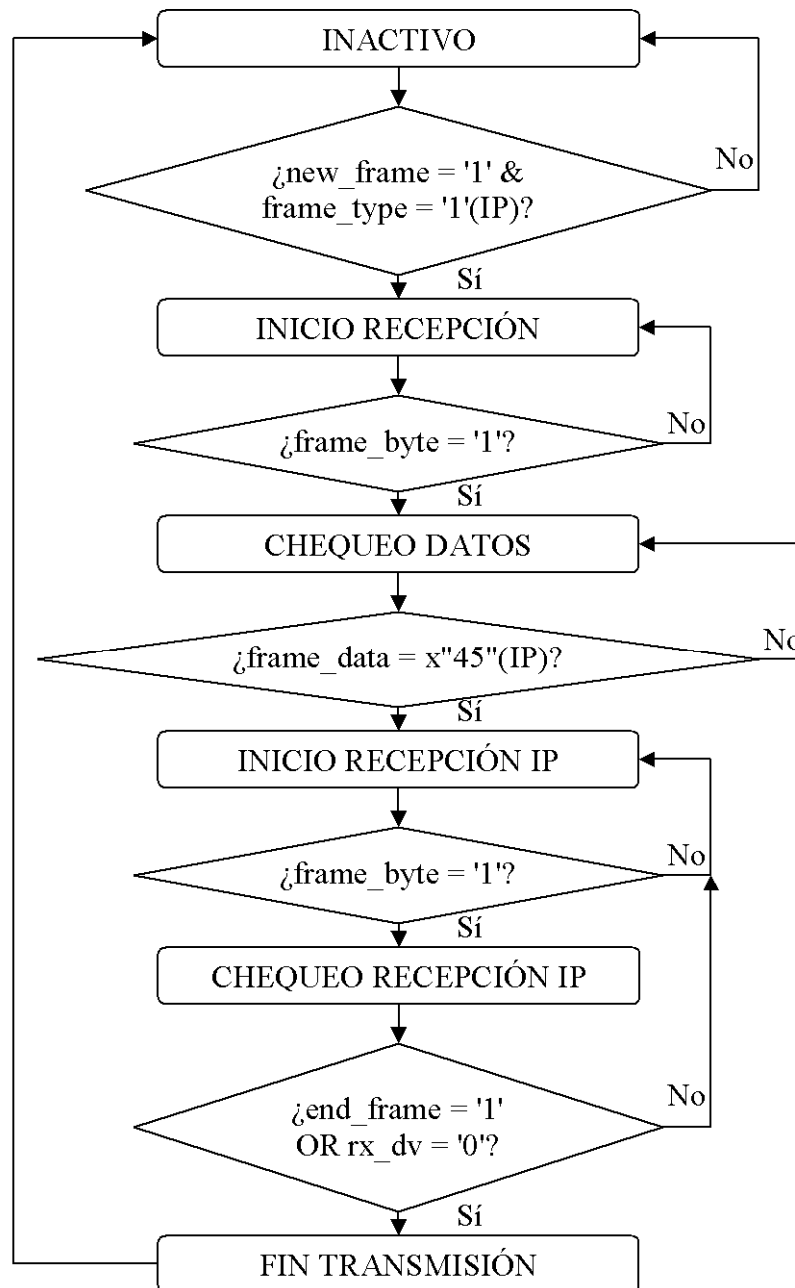


Diagrama 5.1. Identificación de nueva trama, Identificación de paquete IP (x"45" - Versión y Tamaño cabecera) y Recepción de paquete IP.

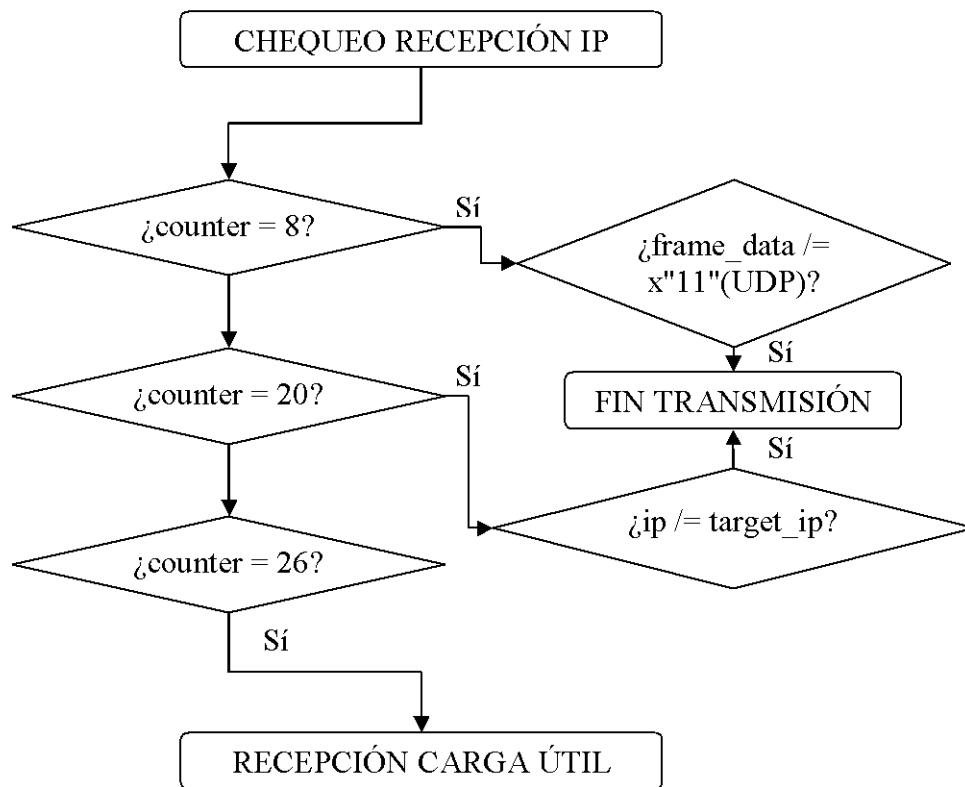


Diagrama 5.2. Identificación de protocolo UDP (x"11"), Identificación de dirección IP y Recepción de carga útil.

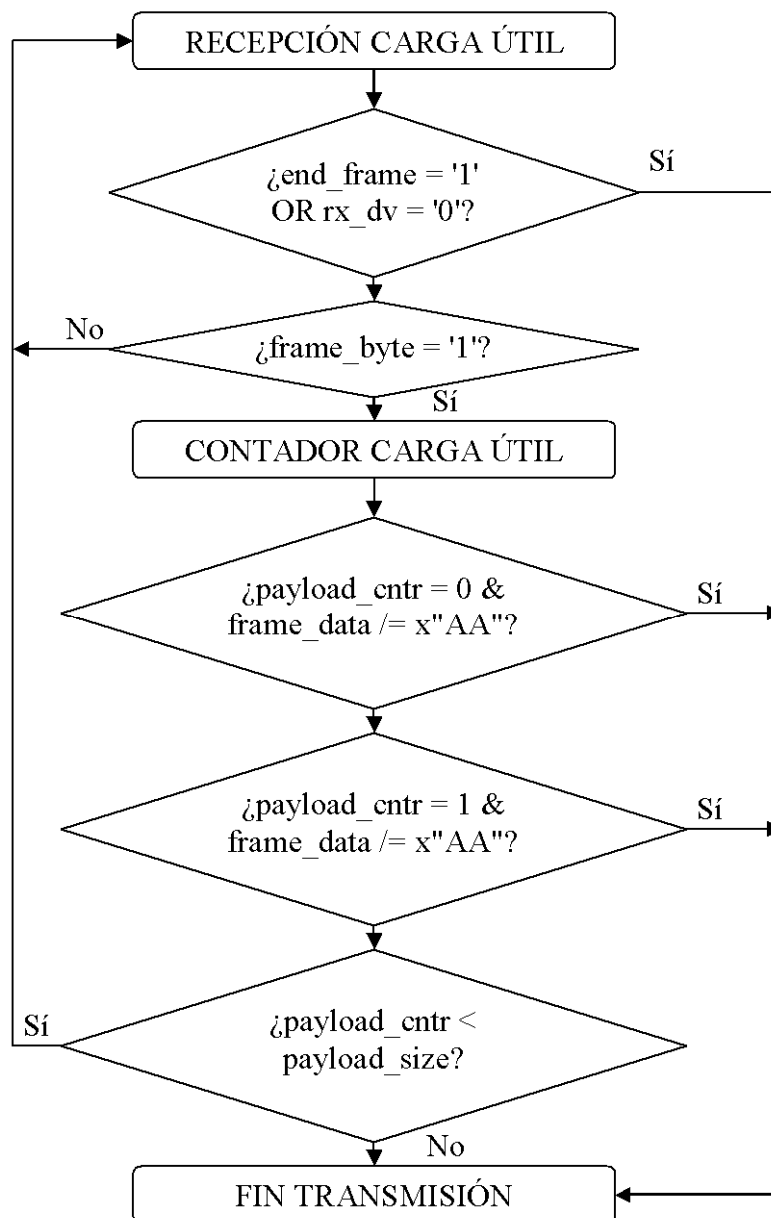
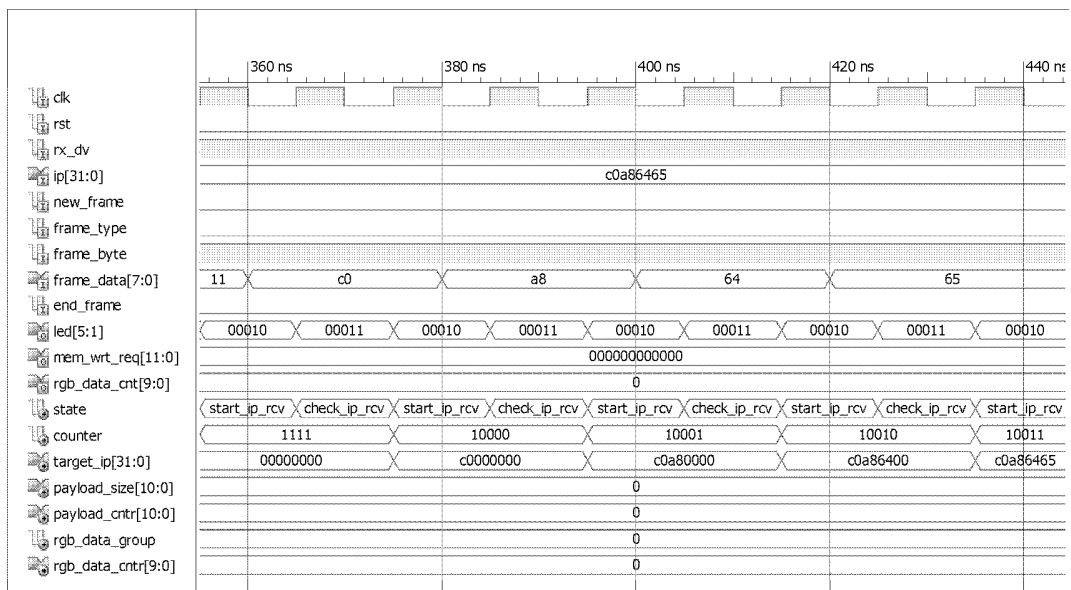
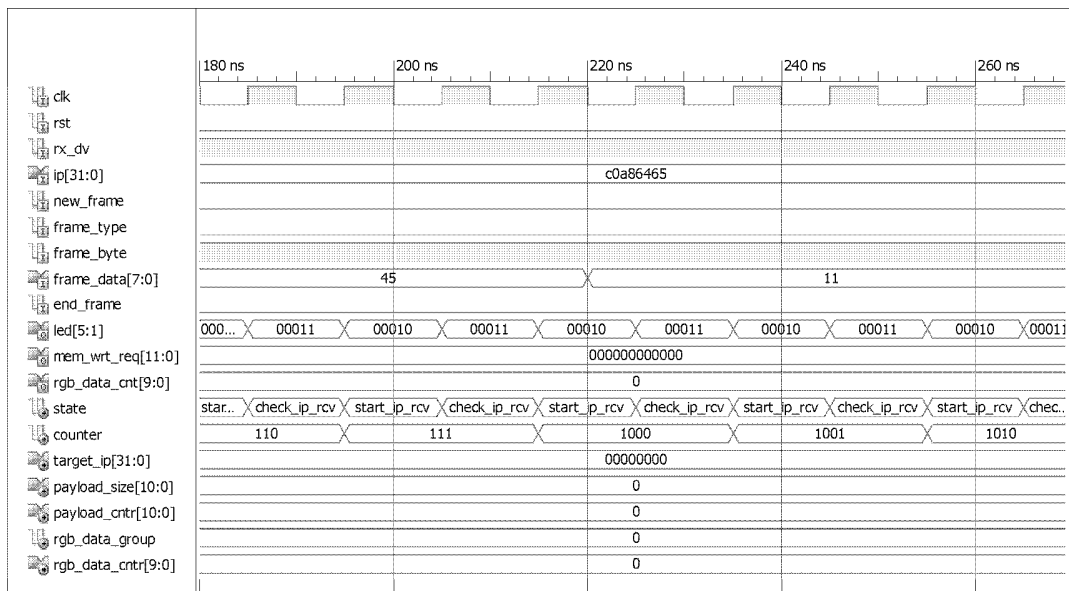
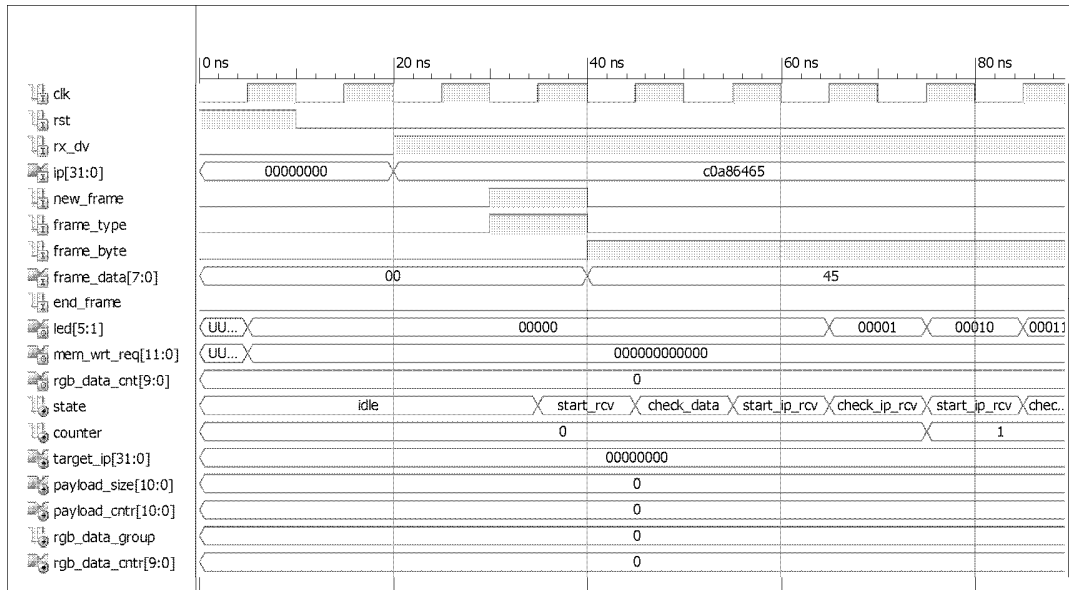
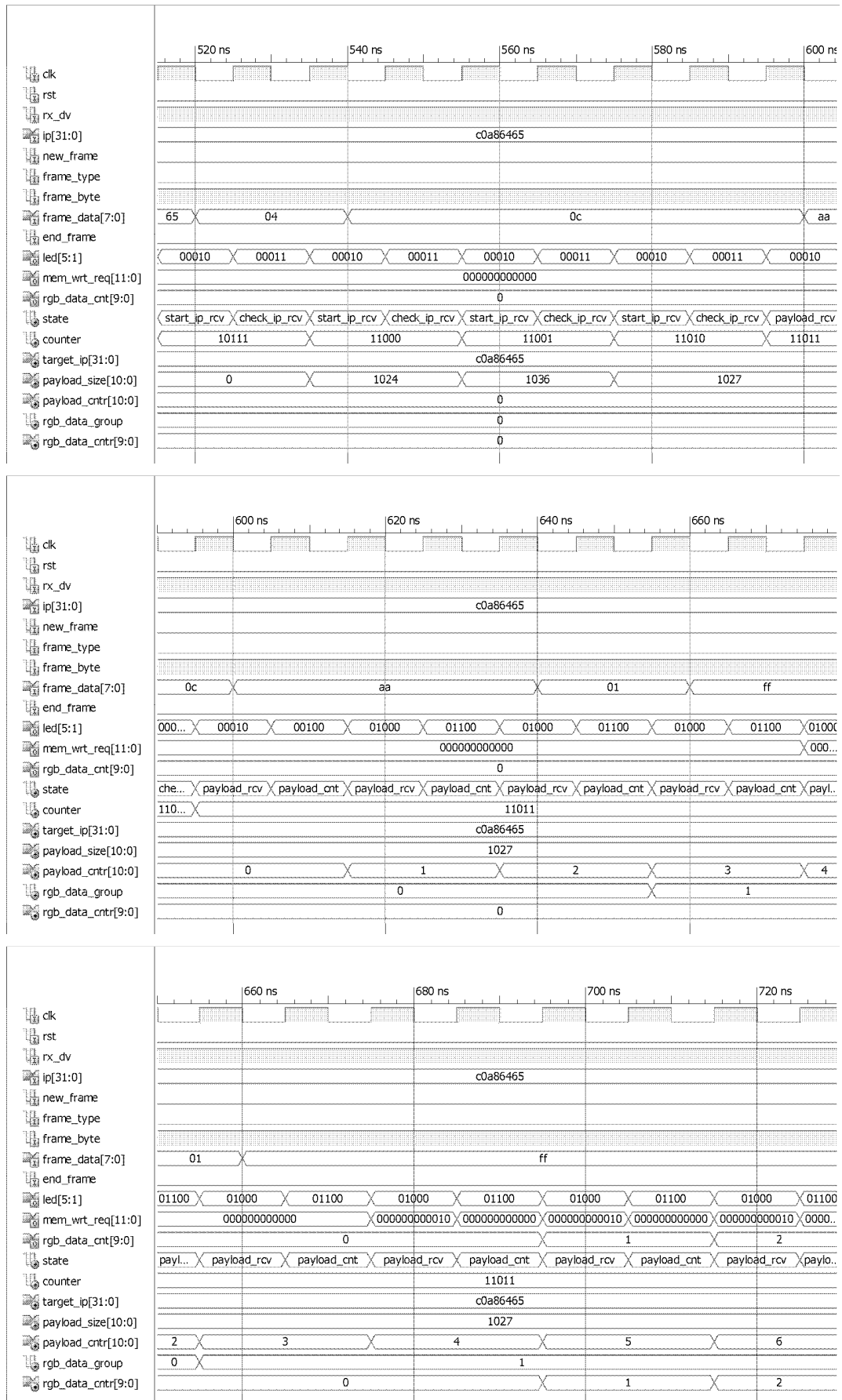


Diagrama 5.3. Identificación de protocolo de vídeo (x"AAAA" - Protocolo personalizado) y Recepción de información de vídeo.

A continuación se muestran los resultados de las simulaciones que demuestran el funcionamiento del módulo 'rx_control', donde se pueden ver todos los estados y transiciones de estados que aparecen en el diagrama de flujo.





5.3.2.3 Módulo 'ram_memory'

El módulo 'ram_memory' implementa una memoria RAM donde se almacenan los datos de imagen recibidos a través del módulo 'rx_control'.

La memoria RAM está compuesta por 12 bloques de memoria de 1024 posiciones de 8 bits cada uno. Cada uno de los bloques de memoria almacena los valores de intensidad para cada componente de color de cada LED y grupo de datos. Es decir, las memorias 1 a 3 para las componentes de color R, G y B del grupo de datos 1, las memorias 4 a 6 para las componentes de color R, G y B del grupo de datos 2, y así sucesivamente.

Los bloques de memoria se pueden leer y escribir a la vez. Las entradas de todos los bloques de memoria están conectadas a la salida de la interfaz Ethernet PHY, siendo el módulo 'rx_control' el encargado de seleccionar el bloque de memoria que se actualiza en cada momento. El módulo 'display_control' lee todas las salidas de los bloques de memoria a la vez, ya que todos los grupos de datos se actualizan a la vez.

En la Figura 5.17 se muestra una representación de uno de los bloques de memoria del módulo 'ram_memory', donde se pueden ver las señales de entrada (izquierda) y salida (derecha) las cuales se listan y comentan a continuación:

Entradas:

- rclk: Reloj para lectura.
- wclk: Reloj para escritura.
- we: Habilitación para escritura.
- addri: Dirección para escritura.
- addro: Dirección para lectura.
- di: Dato de entrada.

Salidas:

- do: Dato de salida.

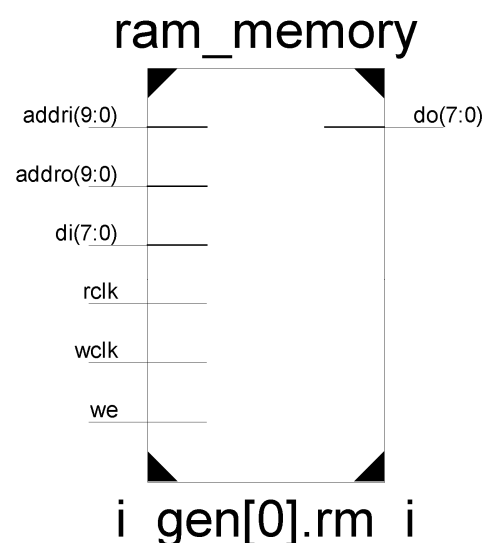


Figura 5.17. Representación de uno de los bloques de memoria del módulo 'ram_memory'.

5.3.2.4 Módulo 'display_control'

El módulo 'display_control' recibe los datos de la memoria RAM y genera todas las señales necesarias para los drivers del módulo LED. El módulo 'display_control' incluye a su vez el módulo 'gamma_table' (Figura 5.19) que consiste en una tabla de búsqueda para realizar la corrección Gamma.

En la Figura 5.18 se muestra una representación del bloque del módulo 'display_control', donde se pueden ver las señales de entrada (izquierda) y salida (derecha) las cuales se listan y comentan a continuación (sólo se muestran las señales que no se han definido en módulos anteriormente descritos):

Entradas:

- display_ena: Habilitación del control del módulo LED.
- ram_data: Dato de 96 bits proveniente de la memoria RAM.

Salidas:

- ram_address: Direccionamiento para la lectura de la memoria RAM.

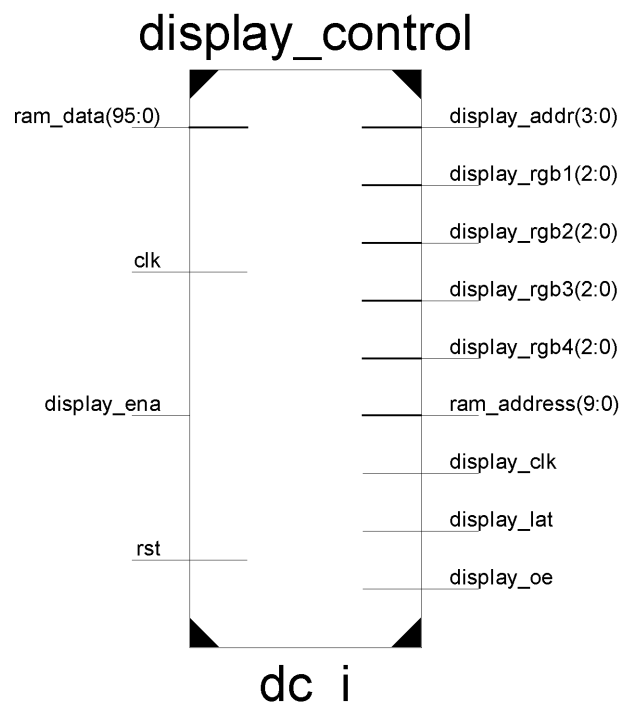


Figura 5.18. Representación del módulo 'display_control'.

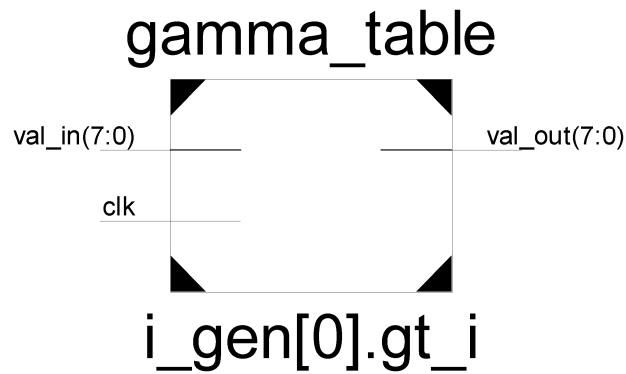


Figura 5.19. Detalle del módulo ‘gamma_table’ incluido en el módulo ‘display_control’.

En el diagrama de flujo que sigue se muestra el funcionamiento de la máquina de estados con la que se implementa el módulo ‘display_control’.

En el diagrama de flujo aparece la lectura en memoria correspondiente a un solo grupo de datos (24 bits). Recordando que todos los grupos de datos se actualizan a la vez, el control mostrado en el diagrama de flujo se realiza en paralelo para los cuatro grupos de datos a la vez.

Con la corrección Gamma se dedican menos bits para codificar los valores más brillantes de la imagen que no somos capaces de diferenciar y más bits para codificar los valores más oscuros de la imagen para los que tenemos más sensibilidad. El nivel de corrección gamma se ha ajustado experimentalmente hasta conseguir una visualización aceptable, quedando establecido dicho valor en 2.8.

Para alcanzar la escala de grises de 256 niveles hay que cargar en los drivers de acceso de línea los niveles de salida del PWM 256 veces por periodo de escaneo. Esto permite que se pueda controlar el nivel de brillo individual de los LEDs en 256 pasos. Por ejemplo, un LED que está encendido en cada uno de los 256 registros de datos estará con brillo máximo. Si el LED está encendido menos de 256 registros estará tenue.

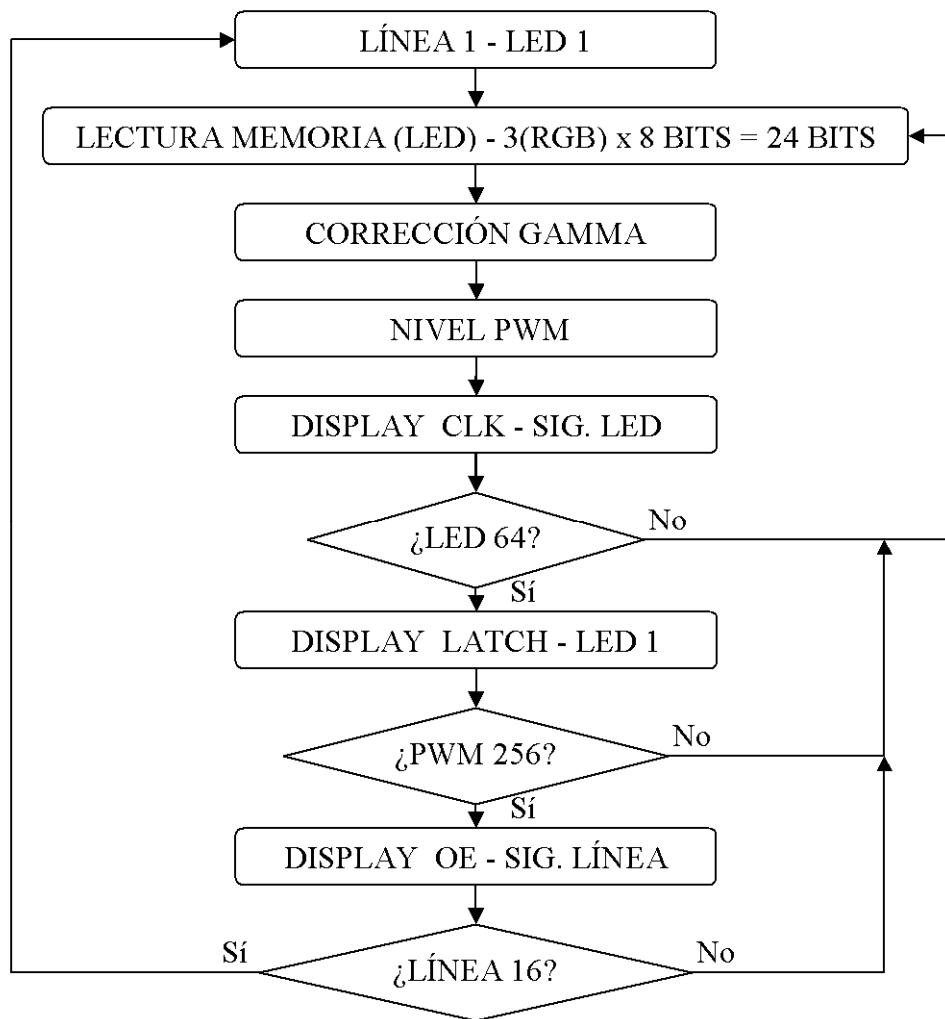
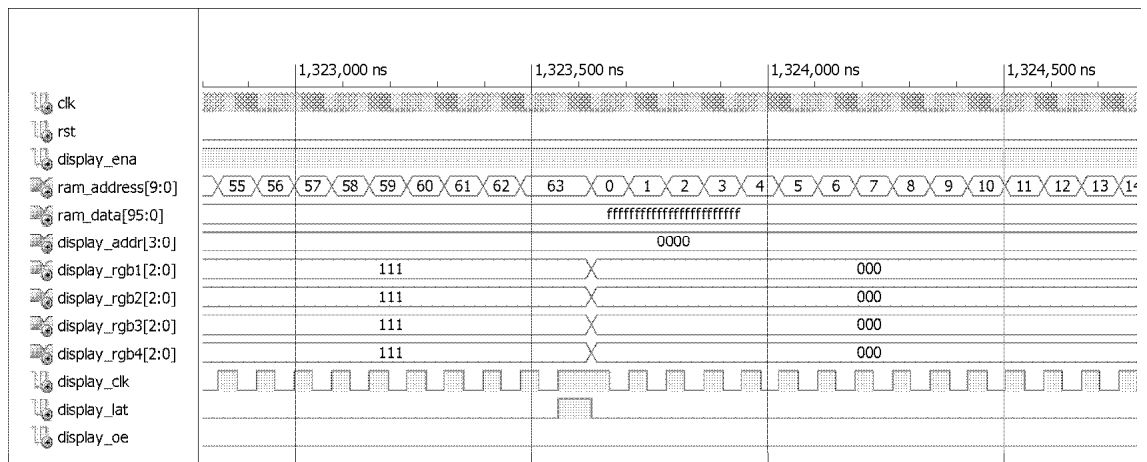
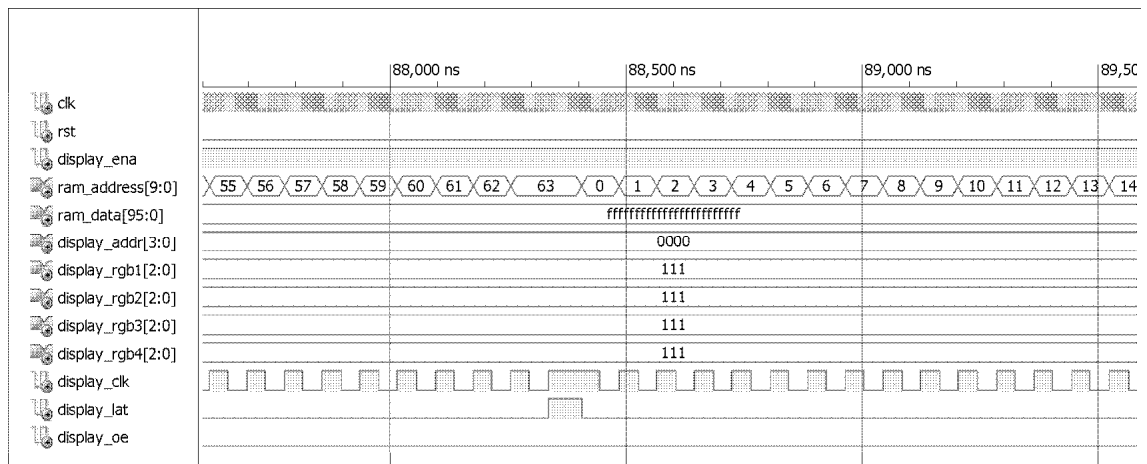
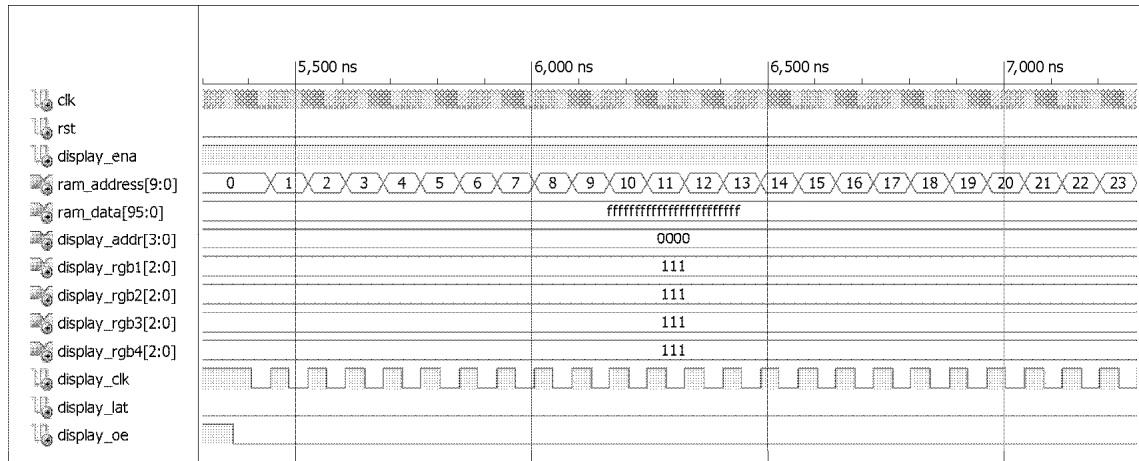
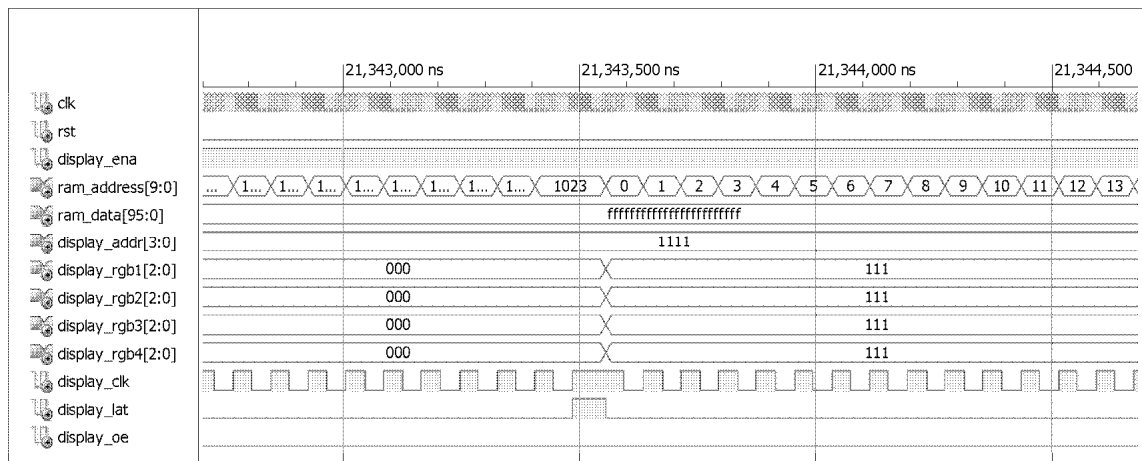
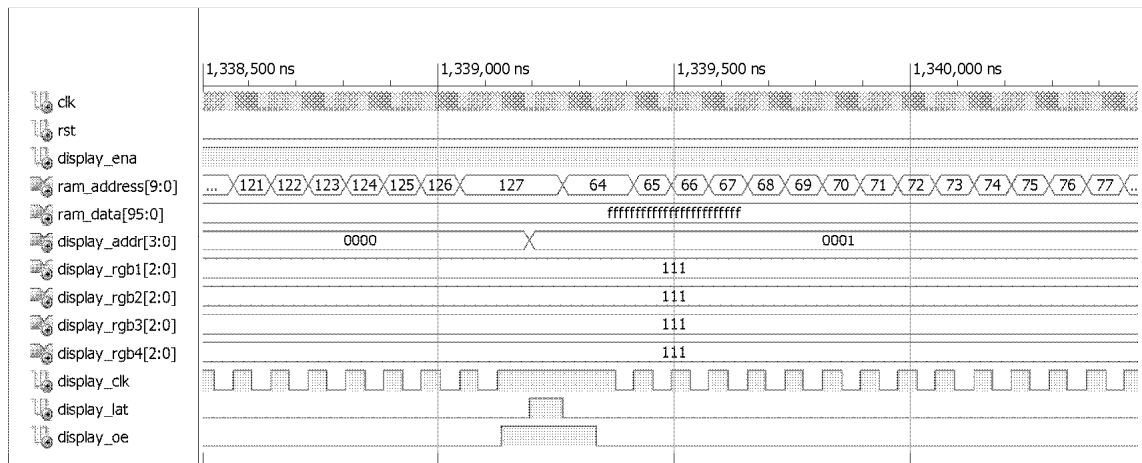
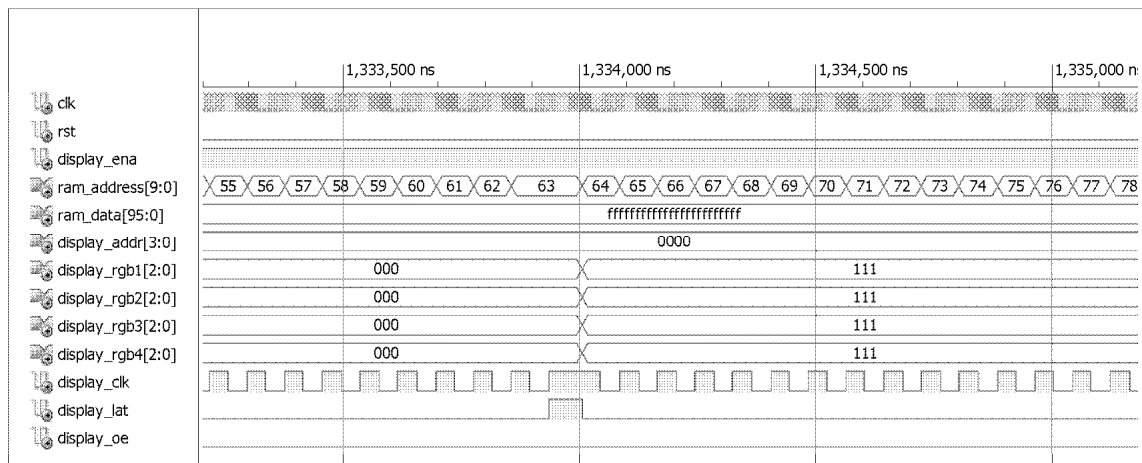


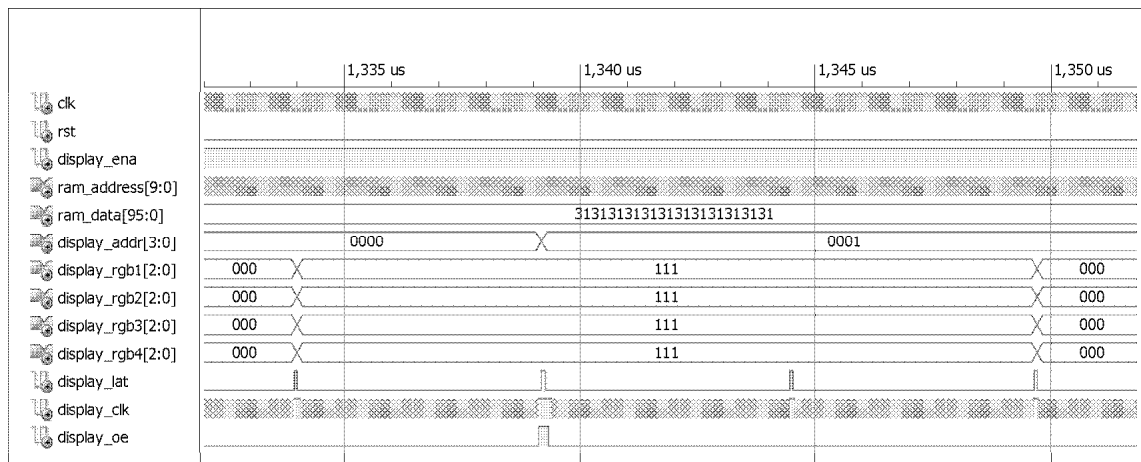
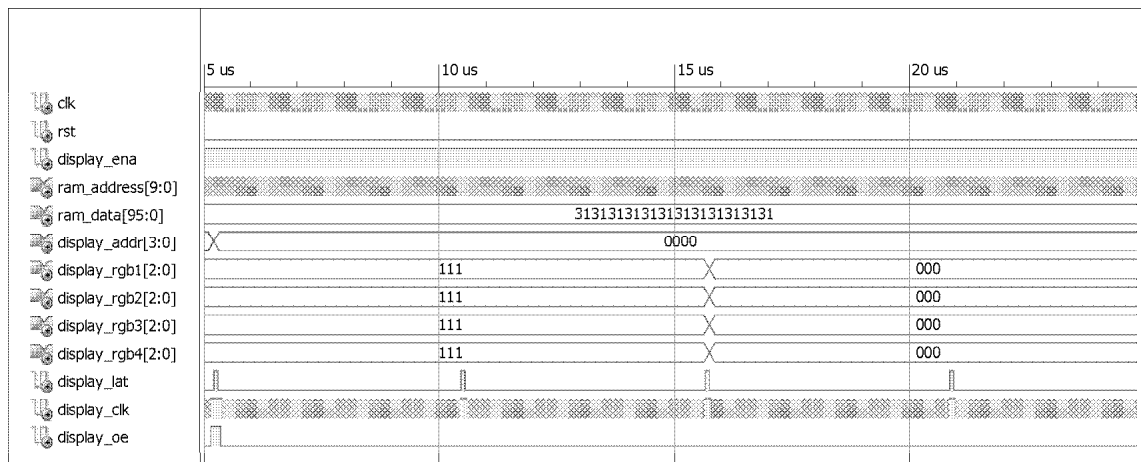
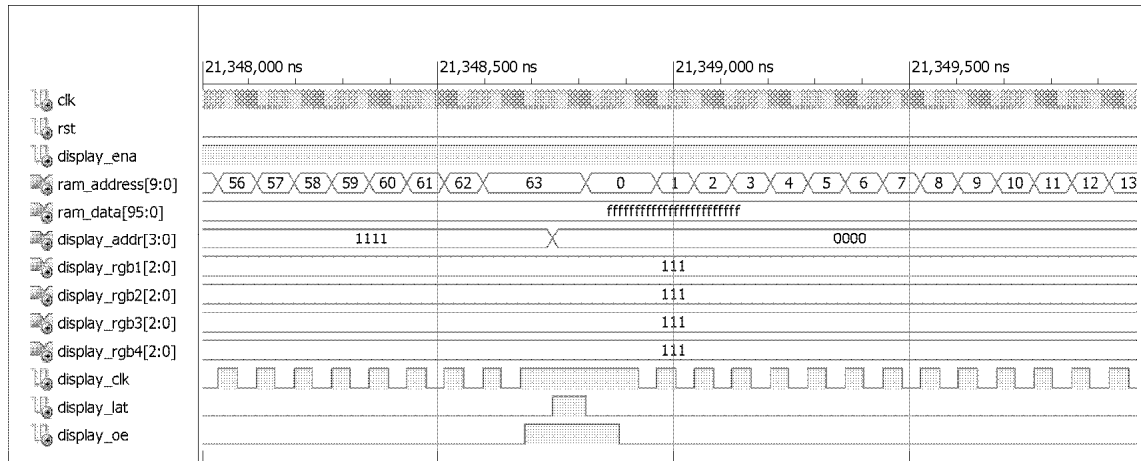
Diagrama 5.4. Máquina de estados del módulo 'display_control'.

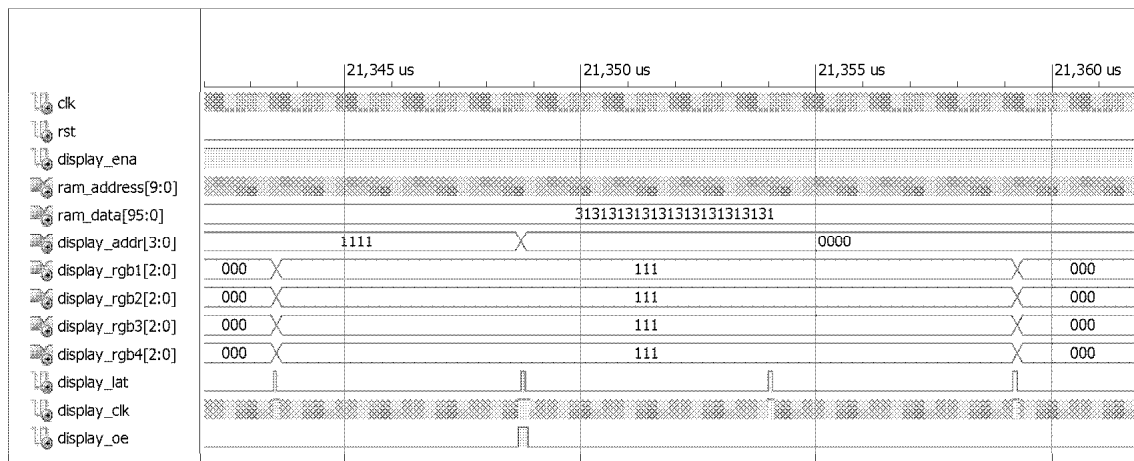
A continuación se muestran los resultados de las simulaciones que demuestran el funcionamiento del módulo 'display_control', donde se puede ver:

- La actualización de la línea 1 para nivel de intensidad máximo.
- El tránsito de las líneas 1 a 2 y 16 a 1 para nivel de intensidad máximo.
- El tránsito de las líneas 1 a 2 y 16 a 1 (con menos zoom) con nivel de intensidad atenuado.









5.3.2.5 Módulo 'dcm1'

El módulo 'dcm1' (DCM - Digital Clock Manager) en nuestro caso se emplea como multiplicador de frecuencia para elevar el reloj del sistema desde los 50 MHz hasta los 120 MHz que admite la FPGA.

En la Figura 5.20 se muestra una representación del bloque del módulo 'dcm1', donde se pueden ver las señales de entrada (izquierda) y salida (derecha) las cuales se listan y comentan a continuación:

Entradas:

- clk_in_in: Señal de reloj proveniente del oscilador de 50 MHz.
- rst_in: Reset.

Salidas:

- clkfx_out: Salida multiplicada en frecuencia (120 MHz).
- locked_out: Enclavamiento del PLL.

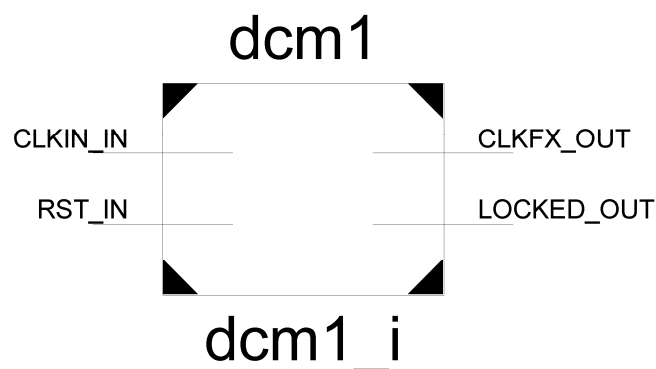


Figura 5.20. Representación del módulo 'dcm1'.

A continuación se muestran las interconexiones entre los módulos ‘ethernet’, ‘rx_control’, ‘ram_memory’ y ‘display_control’.

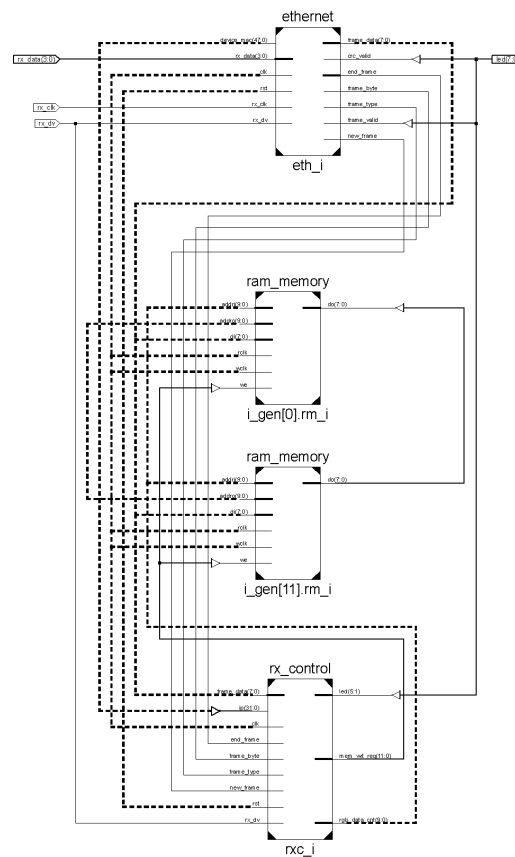


Figura 5.21. Interconexión módulos ‘ethernet’, ‘rx_control’ y ‘ram_memory’.

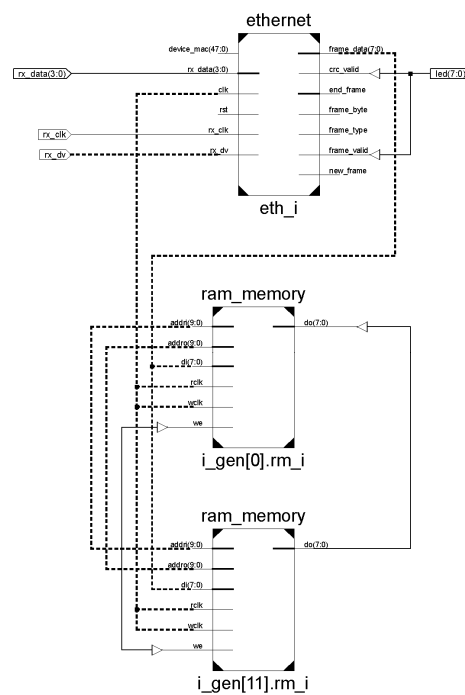


Figura 5.22. Interconexión módulos ‘ethernet’ y ‘ram_memory’.

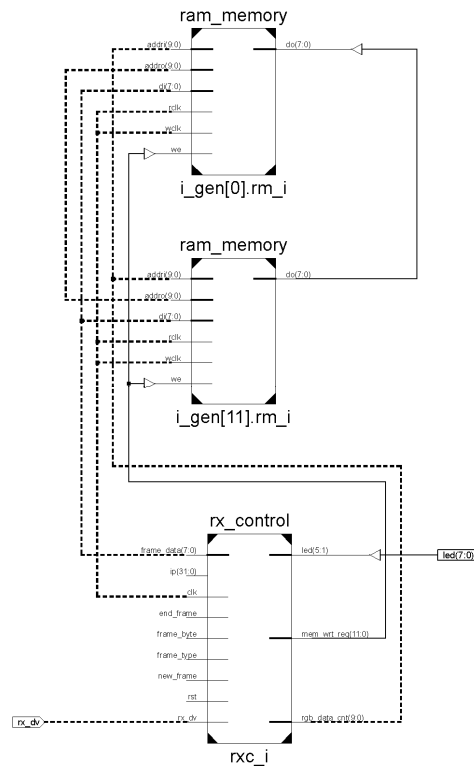


Figura 5.23. Interconexión módulos 'rx_control' y 'ram_memory'.

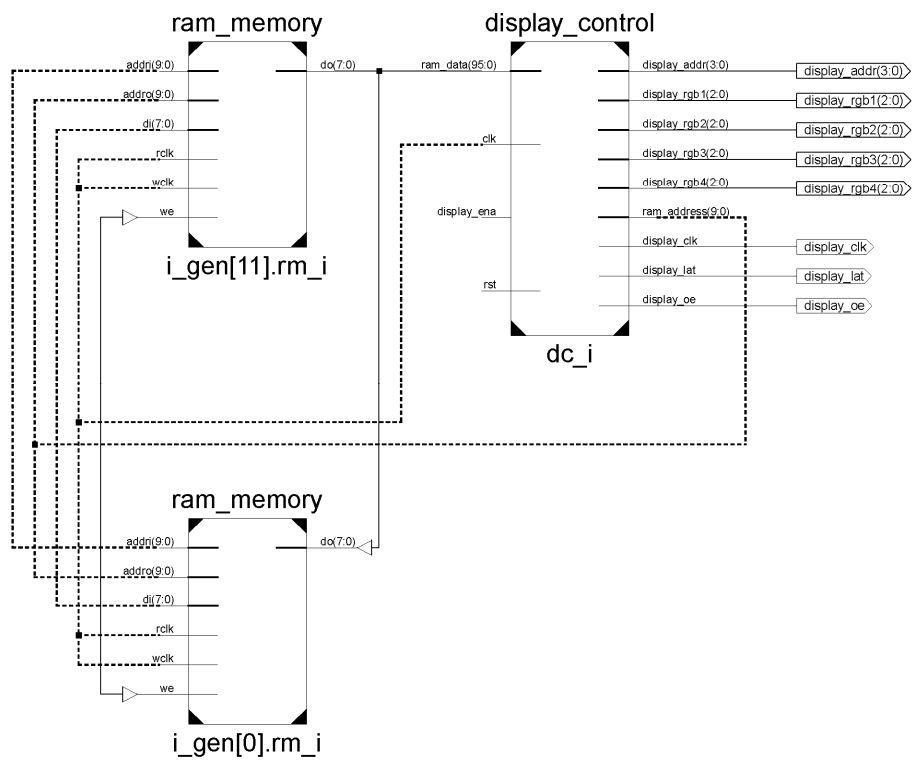


Figura 5.24. Interconexión módulos 'display_control' y 'ram_memory'.

Por último, se muestra un resumen del proyecto desde el entorno de desarrollo ISE de Xilinx, donde se puede observar la jerarquía de los módulos empleados, así como la ausencia de errores, advertencias y restricciones en el diseño.

The screenshot displays the Xilinx ISE Project Navigator window for the project 'fpga_top'. The 'Design Summary' tab is active, showing the project status and device utilization.

fpga_top Project Status (11/20/2017 - 12:49:14)

Project File:	fpga.xise	Parser Errors:	No Errors
Module Name:	fpga_top	Implementation State:	Programming File Generated
Target Device:	xc3s500e-sfg320	Errors:	No Errors
Product Version:	ISE 14.7	Warnings:	No Warnings
Design Goal:	Balanced	Routing Results:	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	Timing Constraints:	All Constraints Met
Environment:	System Settings	Final Timing Score:	0 (Timing Report)

Device Utilization Summary

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	350	9,312	3%	
Number of 4 input LUTs	1,608	9,312	17%	
Number of occupied Slices	890	4,656	19%	
Number of Slices containing only related logic	890	890	100%	
Number of Slices containing unrelated logic	0	890	0%	
Total Number of 4 input LUTs	1,648	9,312	17%	
Number used as logic	1,608			
Number used as a route-thru	40			
Number of bonded IOBs	34	232	14%	
Number of RAMB16s	12	20	60%	
Number of BUFMUXs	1	24	4%	
Number of DCMs	1	4	25%	
Average Fanout of Non-Clock Nets	4.29			

6 APLICACIÓN PARA PC PARA EL ENVÍO DE INFORMACIÓN AL CONTROLADOR DEL MÓDULO LED (FPGA)

Como fuente de información se ha desarrollado una aplicación Visual Basic para PC que permite el envío de patrones de prueba, imágenes y vídeo de baja velocidad.

El interfaz de usuario de la aplicación que se muestra a continuación en la Figura 6.1 consta de los siguientes controles:

- TEST: Enciende el módulo completo con el color seleccionado y el brillo ajustado en el control 'BRIGHTNESS'.
- SCAN: Hace un barrido en intensidad del módulo completo con el color seleccionado desde el brillo mínimo hasta el ajustado en el control 'BRIGHTNESS' y a continuación hasta el brillo mínimo de nuevo. El barrido en intensidad permite detectar fallos en el módulo LED relacionados con la regulación que no son apreciables con brillo máximo.
- TEST H, V y D: Realizan barridos horizontales, verticales y diagonales de líneas blancas y el brillo ajustado en el control 'BRIGHTNESS'. Estos test permiten detectar cortocircuitos y circuitos abiertos en LEDs que no pueden ser detectados si se enciende el módulo completo.
- AUTO: Realiza una secuencia de todos los test programados.
- FRAME: Muestra una imagen preconfigurada con resolución de 64x64 píxeles y formato de mapa de bits (.bmp).
- SEQUENCE: Muestra una secuencia de imágenes preconfiguradas con resolución de 64x64 píxeles y formato de mapa de bits (.bmp).

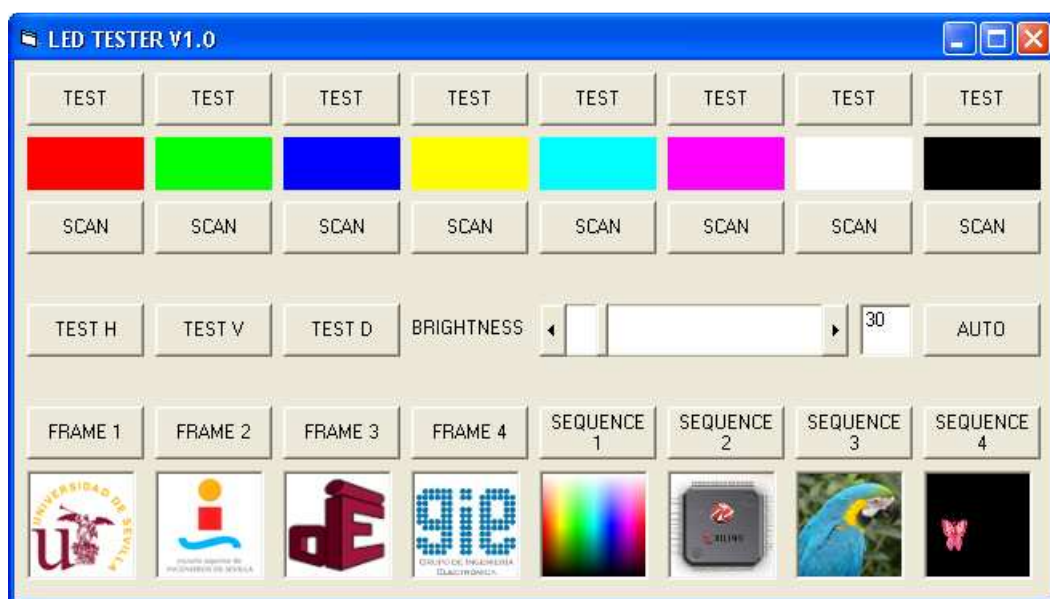


Figura 6.1. Interfaz de usuario de la aplicación VB para PC.

Como en el controlador del módulo LED no está implementado el protocolo ARP (Address Resolution Protocol), que permite obtener su dirección física (Ethernet MAC -> 00-aa-aa-aa-aa-aa) a partir de su dirección IP (192.168.100.101), empleamos el comando 'arp' del Command Prompt de MSDOS para crear la conversión de direcciones de forma manual.

Shell "cmd.exe /k arp -s 192.168.100.101 00-aa-aa-aa-aa-aa", vbHide

Para la actualización completa del módulo LED hay que realizar 12 envíos con el socket de Windows (Winsock). En cada envío se incluyen 1027 bytes, 2 para la cabecera del protocolo de video (x"AAAA"), 1 para la selección de la componente de color y del grupo de datos a actualizar y 1024 para los niveles de gris.

El orden de envío establece también la secuencia de actualización de componentes de color en los grupos de datos, ya que el controlador de escaneo barre la memoria mucho más rápido. Mediante las pruebas realizadas se ha comprobado que la actualización del módulo completo por componentes de color (R1 a 4 -> G1 a 4 -> B1 a 4) produce una sensación de parpadeo menor que si la actualización se realiza por grupos de datos completos (RGB1 -> RGB2 -> RGB3). Se realiza por tanto una actualización por componentes de color tal y como se muestra a continuación con el código del procedimiento de VB encargado de dicha tarea:

Private Sub tx_rgb()

```
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H0) & dg_r(1)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H3) & dg_r(2)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H6) & dg_r(3)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H9) & dg_r(4)

Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H1) & dg_g(1)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H4) & dg_g(2)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H7) & dg_g(3)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&HA) & dg_g(4)

Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H2) & dg_b(1)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H5) & dg_b(2)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&H8) & dg_b(3)
Winsock1.SendData Chr(&HAA) & Chr(&HAA) & Chr(&HB) & dg_b(4)
```

End Sub

Para los test con colores puros o combinaciones de estos basta con enviar una cadena de 1024 caracteres, previamente formateada con el valor de intensidad deseado, para cada una de las componentes de color de cada grupo de datos. A continuación se muestra la generación de las cadenas para intensidad mínima (dg_0), intensidad máxima (dg_255) e intensidad ajustada con el control 'BRIGHTNESS' (dg):

```

dg_0 = ""
dg_255 = ""
For i = 1 To 1024
    dg_0 = dg_0 & Chr(0)
    dg_255 = dg_255 & Chr(255)
Next

text_brt = brt
dg = ""
For i = 1 To 1024
    dg = dg & Chr(brt)
Next

```

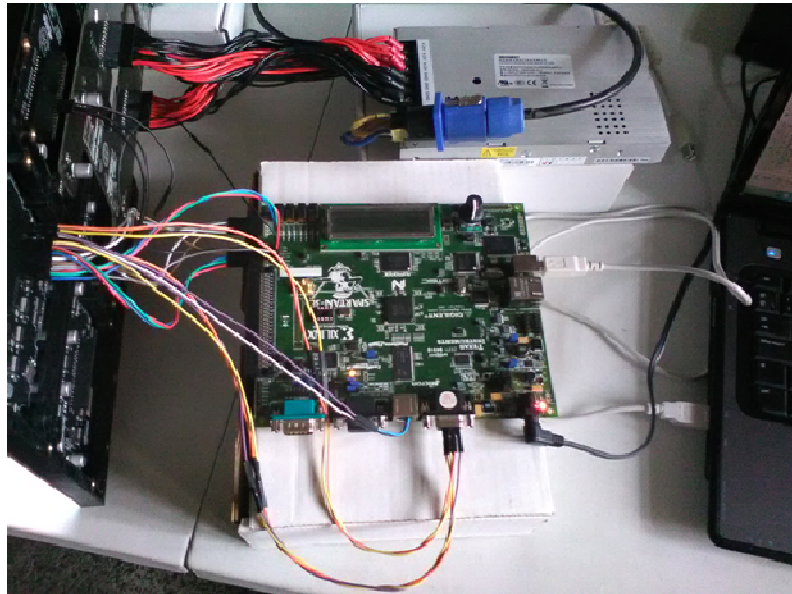
Para los test de líneas horizontales, verticales y diagonales se utilizan una serie de bucles anidados que determinan en cada momento de la secuencia que LEDs de cada fila del grupo de datos han de activarse (se añade a la cadena el valor de intensidad ajustado con el control 'BRIGHTNESS' - dg = dg & Chr(brt)) o han de permanecer desactivados (dg = dg & Chr(0)).

Por último, para los test de imágenes se realizan las siguientes operaciones:

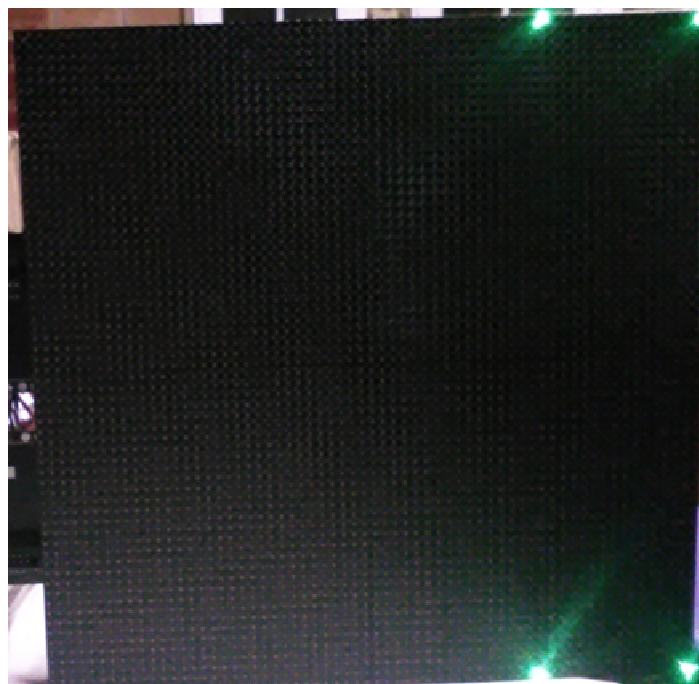
- Se carga la imagen preconfigurada con resolución de 64x64 píxeles y formato de mapa de bits (.bmp).
- Se extrae el alto y ancho de la imagen.
- Se recorre la imagen extrayendo cada pixel (pxl = 'GetPixel').
- Se obtienen las componentes de color para cada pixel:
 - $pxl_r = pxl \bmod 256$
 - $pxl_b = \text{Int}(pxl / 65536)$
 - $pxl_g = (pxl - (pxl_b * 65536) - pxl_r) / 256$
- Se generan las cadenas para cada componente de color y grupo de datos.
- Se llama al procedimiento 'tx_rgb()' para el envío de las cadenas.

7 PRUEBAS Y TEST DEL SISTEMA

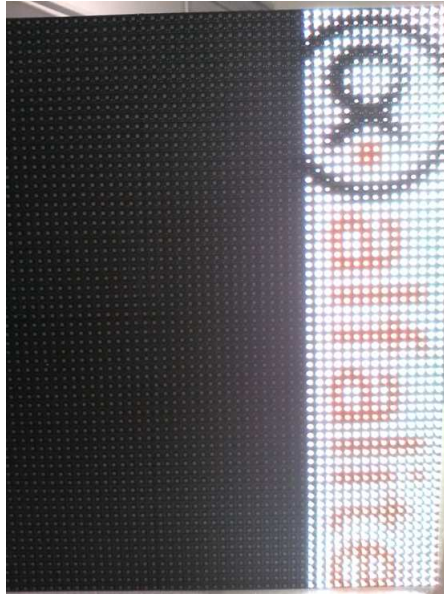
A continuación se muestra una secuencia de imágenes correspondientes a las diferentes pruebas y test que se han realizado durante el diseño y desarrollo del sistema.



1. Sistema compuesto por módulo LED (con su fuente de alimentación), módulo de evaluación para FPGA y PC.



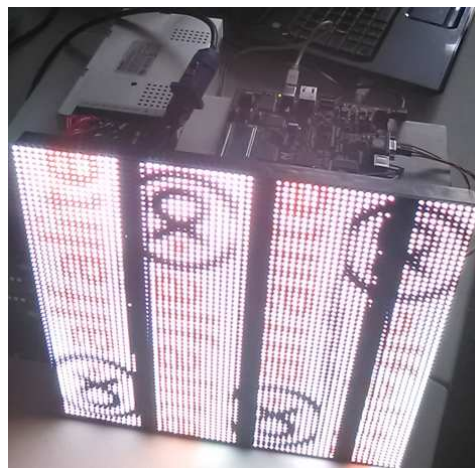
2. Encendido de los LEDs que delimitan el primer grupo de datos para asegurar que se controlan las posiciones y colores deseados.



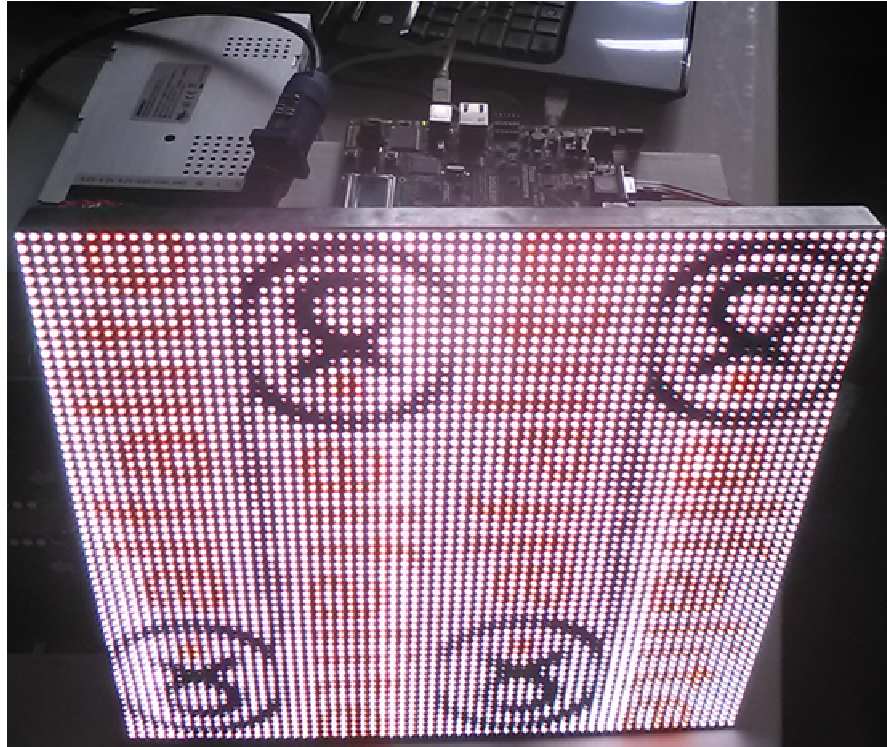
3. Control del primer grupo de datos con volcado desde ROM.



4. Control de los grupos de datos 1 y 2 con volcado desde ROM.



5. Control completo del módulo con volcado desde ROM. Scan con frecuencia de sistema de 50 MHz.



6. Scan mejorado con frecuencia de sistema de 100 MHz.



7. Scan mejorado con frecuencia de sistema de 120 MHz.



```
clc;

clear all;

close all;

img=imread('logo_64_64.bmp');

r=img(:,:,1);

g=img(:,:,2);

b=img(:,:,3);

subplot(2,2,1), imshow(img), title('Original');

subplot(2,2,2), imshow(r), title('Plano R');

subplot(2,2,3), imshow(g), title('Plano G');

subplot(2,2,4), imshow(b), title('Plano B');

fid=fopen('logo_64_64.txt','wt');

i=0;

for j=1:16

    for k=1:64

        fprintf(fid,'%d => x"%02X" & x"%02X" & x"%02X" & x"%02X" & x"%02X" & x"%02X" & x"%02X" & x"%02X" & x"%02X" & x"%02X" & x"%02X"',..

            ,i,b(j+48,k),g(j+48,k),r(j+48,k),b(j+32,k),g(j+32,k),r(j+32,k),b(j+16,k),g(j+16,k),r(j+16,k),b(j,k),g(j,k),r(j,k));

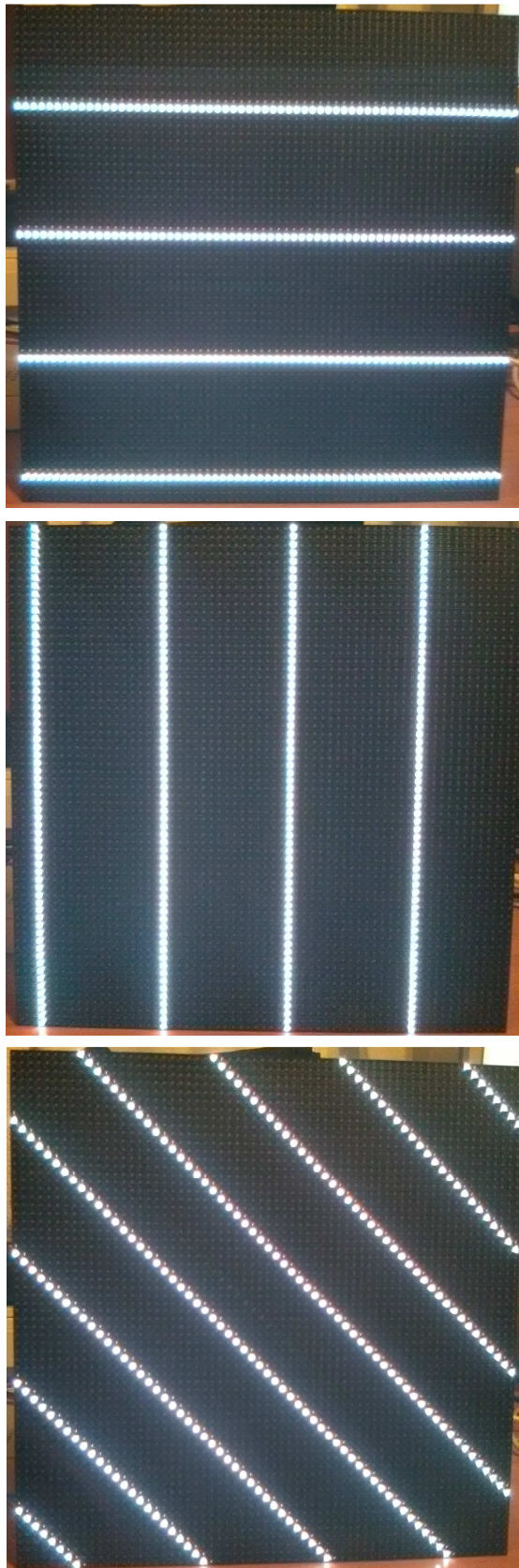
        fprintf(fid,'\n');

        i=i+1;

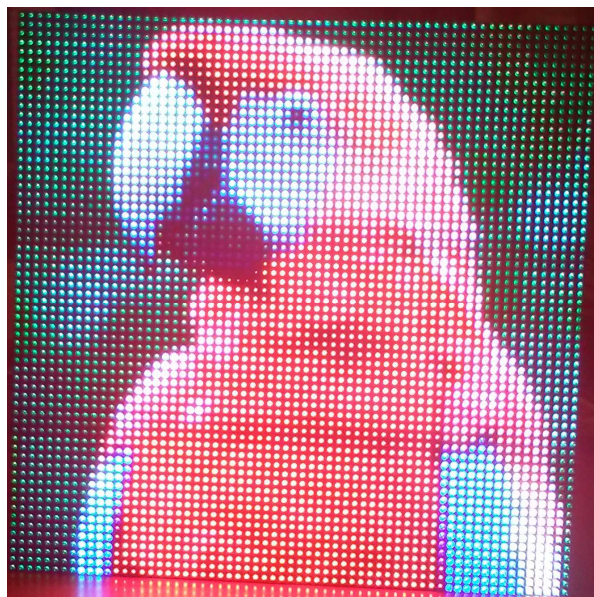
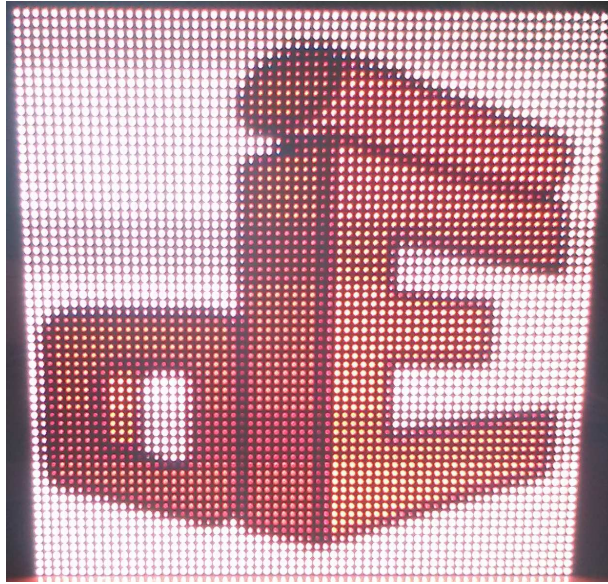
    end

end

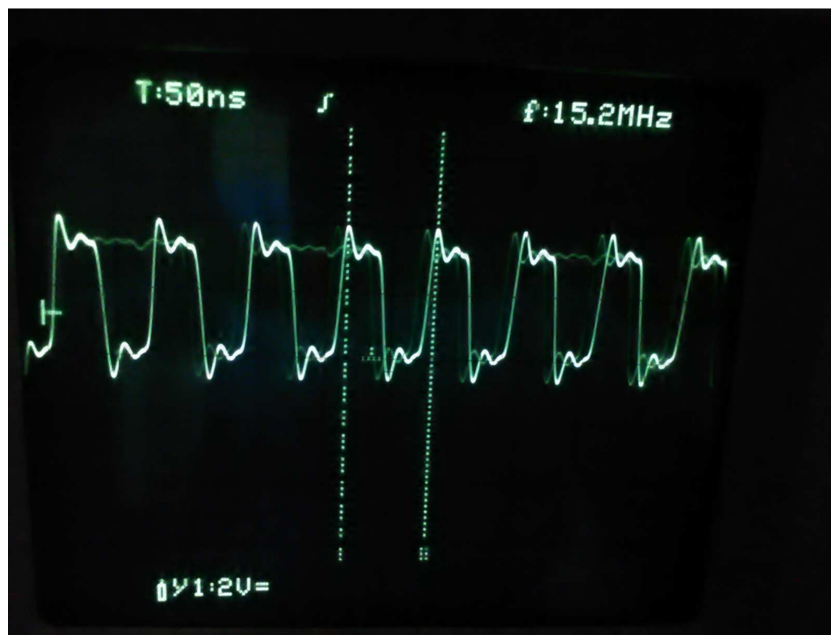
fclose(fid);
```

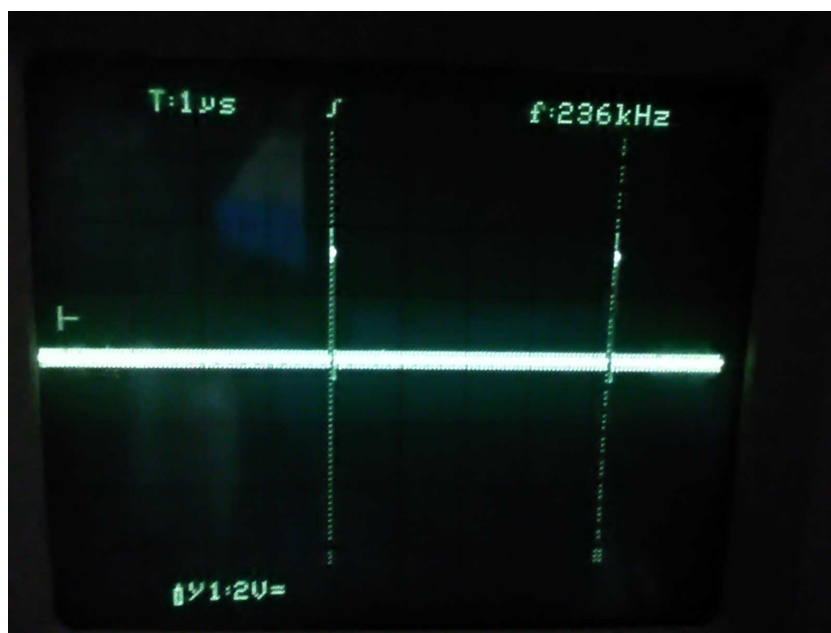
10. Patrones de prueba H, V y D.



11. Frames DIE, GIE y Parrot.



12. Señal de reloj de driver.



13. Señal de latch de driver.

Frecuencia de latch / resolución pwm (256) / scan (16) = Frecuencia actualización grupo de datos = Frecuencia actualización módulo = Frecuencia actualización panel = 57,62 Hz, aproximadamente los 60 Hz exigidos en la especificación de diseño.

8 CONCLUSIONES Y TRABAJOS FUTUROS

Como conclusiones podemos decir que se han conseguido todos los objetivos de diseño, es decir, un controlador para módulo LED, con corrección Gamma, escala de grises de 256 niveles y frecuencia de actualización de pantalla de 60 Hz, capaz de recibir imágenes o patrones de prueba desde una aplicación para PC a través de un interfaz Ethernet.

Se dispone por tanto de un controlador para módulo LED que permite realizar cualquier tipo de prueba que se quiera programar, con un control total y al más bajo nivel de todas las señales.

El controlador se puede utilizar para realizar pruebas de prestaciones, validación y calidad en laboratorio, fábrica y taller. El controlador se puede utilizar también como prototipo de un sistema de bajo coste para imágenes estáticas o de baja velocidad de actualización.

Como trabajos futuros se podrían enumerar los siguientes:

- Elevar la frecuencia de actualización de pantalla utilizando una FPGA que admita una mayor velocidad de operación.
- Extender el control a un panel completo (4 módulos) mediante el empleo de un módulo de evaluación que permita el acceso a más salidas de la FPGA.
- Ampliar el protocolo de la capa de aplicación para que se permita el control de una pantalla completa (varios paneles).
- Utilizar IP Cores para compresión de vídeo y/o comunicación Fast/Gigabit Ethernet que permita la reproducción de vídeo fluido y de alta resolución.
- Modificar el controlador para que permita integrarse en un sistema de inspección automática con visión artificial.

9 REFERENCIAS

- www.alfalite.com
- www.novastar-led.com
- www.broadcom.com
- Avago Technologies (Application Notes)
- TELKOMNIKA, Vol. 11, No. 5, May 2013, pp. 2626 ~ 2633
The Realization of LED Display System Based on the Embedded
- www.mblock.com.tw (Application Notes)
MBI5024 Datasheet VA 02-EN.pdf
- ST (Application Notes)
- Texas Instruments (Application Notes)
- Spartan-3E Start Kit Board User Guide UG230 (v1.0) March 9, 2006
- www.xilinx.com